

ARTIE

ERASMUS+ ARTIE
ARTIFICIAL INTELLIGENCE IN EDUCATION

Izazovi i mogućnosti novog doba
- priručnik za učenike mlađeg uzrasta



Sufinancirano sredstvima
programa Europske unije
Erasmus+

ARTIE: Umjetna inteligencija u obrazovanju - izazovi i prilike novog doba: razvoj novog kurikuluma, vodiča za učitelje i online
tečaja za učenike

Sufinancirano sredstvima programa Europske unije Erasmus+
2020-1-HR01-KA201-077800

ARTIE

Izazovi i mogućnosti novog doba: priručnik za učenike mlađeg uzrasta

Autori

Ivana Ružić
Jura Cmrečak

Izdavač

I. osnovna škola Čakovec, Croatia

“Artificial Intelligence in Education - challenges and opportunities of the new era: development of curriculum, guides for teachers and e-courses for students”, project number 2020-1-HR01-KA201-077800 under Erasmus+ Programme.

Suradnici

Katarzyna Garbacik
Andrzej Garbacik
Bogusław Klimczuk
Željko Krnjajić
Janko Radigović
Ana Pina
Christina Eirini Karvouna

Grafički dizajn i ilustracije

Christina Eirini Karvouna

Prevoditelji

Jura Cmrečak (hrvatski jezik)
Bogusława Denys (poljski jezik)
Ana Pina (portugalski jezik)
Christina Eirini Karvouna (nizozemski jezik)





8

3

Što je umjetna inteligencija?

Na koje načine umjetnu inteligenciju koristimo u svakodnevnom životu?

15

11

Zaigramo UI Bingo!

Na koje načine računala uče i rješavaju probleme?

19

Kako algoritmi oblikuju naš svijet?

23

Kako računala vide svijet oko sebe?

27

Može li računalo prepoznavati poze?

29

Kako etično koristiti podatke za uvježbavanje modela?

36

31

Stvorimo vlastiti UI projekt pomoću Scratcha!

Zabavimo se uz umjetnu inteligenciju!

40

38

Reciklirajmo zajedno!

Čavrljajmo!

45

43

Pametni dom

Upoznajmo Codey Rocky-a

55

51

Codey Rocky upoznaje slijed naredbi i petlje

Codey Rocky vozi sigurno i odgovorno

65

60

Codey Rocky upoznaje funkcije

Kamen, škare i papir!





Što je umjetna inteligencija?

Mogu li računala ili drugi pametni uređaji razmišljati ili rješavati probleme?

Što je umjetna inteligencija?

Koristimo li umjetnu inteligenciju? Kada? Kako? Gdje?

U nasdtavku ćemo naučiti što je umjetna inteligencija i gdje je koristimo u svakodnevnom životu.





Što je umjetna inteligencija? Koja je njena definicija?
Od kada ona postoji?
Mogu li uređaji biti pametni/inteligentni?
Mogu li uređaji zamijeniti ljude?
Što je idealan inteligentan stroj?
Koje su pozitivne i negativne strane umjetne inteligencije?



Umjetna inteligencija (UI)

Umjetna inteligencija je mogućnost računalnog programa ili stroja da razmišlja i uči. To je ujedno i znanstveno polje koje nastoji napraviti računala „pametnima“. Taj pojam je osmislio John McCarthy u 1955.

Pojam „umjetna inteligencije“ denotativno označava stroj koji oponaša ljudsko razmišljanje (kogniciju). U najmanju ruku tu spada mogućnost računala da uči i rješava probleme, ali na drugačiji način od ljudi.





Idealan (savršen) pametni stroj je fleksibilan agent koji percipira svoje okruženje i poduzima radnje koje maksimalno povisuju šansu postizanja nekog cilja. Kako strojevi postaju sposobniji, mentalne sposobnosti za koje se jednom smatralo da zahtijevaju inteligenciju se danas više ne smatraju dijelom umjetne inteligencije. Na primjer, optičko prepoznavanje slova ne smatramo više umjetnom inteligencijom već rutinskom tehnologijom.

Trenutno se pojam umjetne inteligencije koristi za uspješno prepoznavanje ljudskog govora, natjecanjima u kompliciranim strateškim igrama (npr. šah), samovozeće automobile i interpretaciju kompleksnih podataka. Neki ljudi smatraju da ovakav nezaustavni razvoj umjetne inteligencije predstavlja opasnost čovječanstvu.

Ekstreman cilj istraživanja umjetne inteligencije je stvoriti program koji može učiti, rješavati probleme i logički razmišljati. U praksi, aplikacije ipak biraju probleme koje računala mogu lako riješiti. Pretraživanje baza podataka i izračunavanje su radnje koje računala rade bolje od ljudi. S druge strane, percipiranje vlastite okoline, kreativnost i sl. su vještine u kojima su ljudi neusporedivo uspješniji od računala.

Umjetna inteligencija uključuje više područja znanosti poput računalne znanosti, matematike, lingvistike, psihologije, neuroznanosti i filozofije. Istraživači nastoje dostići „generalnu umjetnu inteligenciju“ koja može riješiti više problema, a ne fokusirati se samo na jedan. Istraživači također nastoje stvoriti umjetnu inteligenciju koja je kreativna i osjećajna, koja bi mogla osjećati empatiju ili stvarati umjetnost. Za sada se tome pristupalo na različite načine s brojnim alatima.

Povijest umjetne inteligencije

Ideja „stroja koji razmišlja“ datira iz antičke Grčke. Od razvoja električnog računalstva (i relativno nekim temama spomenutim u ovom članku) važni događaji i prekretnice u evoluciji umjetne inteligencije su:

1950: Alan Turing objavljuje „Computing Machinery and Intelligence“. U ovom znanstvenom radu, Turing, koji je poznat po dekodiranju nacističkog ENIGMA koda, predlaže odgovor na pitanje: „mogu li strojevi razmišljati?“ i predstavlja Turingov Test. Ovaj test determinira ako računalo može demonstrirati istu inteligenciju (ili rezultate inteligencije) kao čovjek. Vrijednost ovog testa je predmet debata i danas.



1956: John McCarthy je osmislio i predstavio pojam „umjetna inteligencija“ na prvoj konferenciju o umjetnoj inteligenciji na Sveučilištu Dartmouth (on je izumio i Lisp jezik). Kasnije te godine, Akken Newell, J.C. Shaw i Herbert Simon su kreirali prvi računalni program temeljen na umjetnoj inteligenciji kojem su dali naziv „Logic Theorist“.

1967: Frank Rosenblatt je sagradio Mark 1 Perceptron, prvu neuralnu računalnu mrežu koja je „učila“ na metodi pokušaja i pogreške. Kasnije u istoj godini, Marvin Minsky i Seymour Papert objavlju knjigu pod naslovom „Perceptrons“, koja je postavila temelje za rad na neuralnim mrežama i na kratko služila kao argument protiv daljnjih istraživanja neuralnih mreža.

The 1980s: Neuralne mreže koje koriste poseban algoritam kako bi se uvježbale i postale široko korištene u aplikacijama koje koriste umjetnu inteligenciju.

1997: IBM-ov Deep Blue pobjeđuje svjetskog prvaka Garryja Kasparovog u šahu (i pobjeđuje u ponovnom susretu).

2011: IBM-ov Watson pobjeđuje Kena Jenningsa and Brada Ruttera u igri „Jeopardy!“.

2015: Baidu-ov Minwa superračunalo koristi poseban tip duboke neuralne mreže naziva „convolutional neural network“ kako bi identificirao i kategorizirao slike s većom stopom preciznosti nego prosječan čovjek.

2016: DeepMind kreira AlphaGo program kojeg pokreće duboka neuralna mreža te tim programom pobjeđuje Leea Sedola, svjetskog prvaka u igri „Go“. Program ga je pobijedio u igri od pet mečeva. Pobjeda je važna zbog velikog broja mogućih pokreta koje igra ima (preko 14.5 trilijuna nakon četiri pokreta). Google je navodno kupio DeepMind za 400 milijuna dolara.





Praktičan rad:

Prepoznamo i istaknimo prednosti i mane umjetne inteligencije.
Izradimo zajednički online poster (s pomoću aplikacija poput Padlet, Lino).

Crtežom prikazimo razvoj umjetne inteligencije.
Crtežom prikazimo vlastita predviđanja razvoja umjetne inteligencije u budućnosti.

Osmislimo i stvorimo primjer umjetne inteligencije koja im pomaže u svakodnevnim aktivnostima (obavljanje zadataka u školi i kod kuće, za hobije ili slobodno vrijeme).
Koje su karakteristike zamišljenog proizvoda/usluge?
Kako će olakšati zadatke?
Zašto je posebno i kako će privući buduće kupce?

Organizirajmo natjecanje koristeći besplatne online alate koji koriste umjetnu inteligenciju:
<https://studio.code.org/s/oceans/lessons/1/levels/1>
https://petalica-paint.pixiv.dev/index_en.html

ZAKLJUČAK

*Umjetna inteligencija je računalni sustav koji se temelji na racionalnosti i razmišljanju, a ne djelovanju.
Ljudski pristup: sustavi koji razmišljaju i djeluju kao ljudi.
Idealan pristup: sustavi koji racionalno razmišljaju i racionalno djeluju.*

Promisli!



Što sada znam?



Što sam naučio/la?



Što još želim znati?



Na koje načine umjetnu inteligenciju koristimo u svakodnevnom životu?

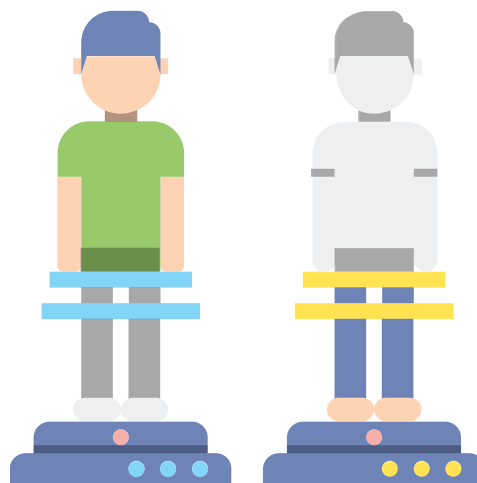
Mogu li računala ili drugi pametni uređaji razmišljati ili rješavati probleme? Ako da, koje?

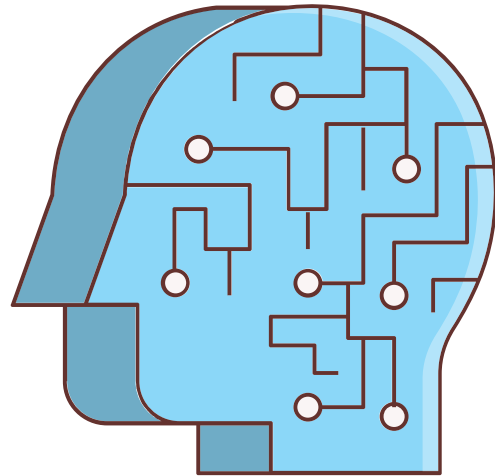
Koristimo li umjetnu inteligenciju u svakodnevnom životu?

Kako koristimo umjetnu inteligenciju? Kada? Gdje?

Možemo li koristiti umjetnu inteligenciju za učenje? Na koje načine?

U nastavku ćemo učiti o umjetnoj inteligenciji i kako je koristimo u svakodnevnom životu.





Teme za raspravu:

Koristimo li umjetnu inteligenciju? Kada?

Primjeri za raspravu mogu biti „face ID“, Google pretraživač, društvene mreže, preporuke za šoping, digitalni pomoćnici, Netflix, pametni kućni uređaji, itd.

Može li se umjetna inteligencija zloupotrijebiti? Kako?

Primjena umjetne inteligencije



Umjetna inteligencija se danas primjenjuje na brojne načine. Pogledjamo najčešće primjere.

Prepoznavanje govora: Ono je poznato kao automatsko prepoznavanje govora, računalno prepoznavanje govora ili „speech-to-text“. Računalno prepoznavanje ima mogućnost koja koristi prirodno prepoznavanje govora kako bi se ljudski govor procesuirao u pisani format. Brojni mobilni uređaji imaju prepoznavanje govora ugrađeno u svoje sustave koji omogućuju glasovno pretraživanje (npr. Siri) i olakšani pristup za slanje poruka.

Služba za korisnike: Online chatbotovi zamjenjuju ljudske agente/pomoćnike u svijetu kupnje. Odgovaraju na često postavljena pitanja (FAQs) za teme poput otpreme, pružaju mogućnosti personaliziranih savjeta, unakrsno prodaju proizvode, predlažu veličine i time mijenjaju način razmišljanja o angažmanu prema kupcu na mrežnim stranicama i društvenim mrežama.

Računalni vid: Ova tehnologija umjetne inteligencije omogućuje računalima i drugim sustavima da iz digitalnih slika, videa i drugih vizualnih podataka prepoznaju i izdvajaju značajne informacije te na temelju tih informacija izvršavaju određene radnje. Ova mogućnost davanja preporuka se razlikuje od zadataka prepoznavanja slike. Računalni vid se temelji na složenim neuralnim mrežama te se koristi u „tagiranju“ na društvenim mrežama, radiologiji i samovozećim automobilima u automobilskoj industriji.



Sustavi za preporučivanje: Korištenjem podataka o prošlim potrošnjama, algoritam na bazi UI-je može otkriti trendove koji mogu biti korišteni za efikasnije strategije o unakсноj prodaji.

Automatsko trgovanje dionicama: Dizajnirano s ciljem optimizacije portfolija dionica, ova platforma za trgovanje dionicama na bazi umjetne inteligencije automatski trguje nekoliko tisuća ili čak milijuna puta bez ljudske intervencije.

Praktičan rad:

Zamislimo i stvorimo primjer umjetne inteligencije koja nam pomaže u svakodnevnim aktivnostima (obavljanje zadataka u školi i kod kuće, za hobije ili slobodno vrijeme).

Koje su karakteristike zamišljenog proizvoda/usluge?

Kako će olakšati zadatke?

Zašto je posebno i kako će privući buduće kupce?

Interaktivni zadaci:

Pronađi svoj put od škole do željene destinacije!

Koristi Google karte kako bi istražio i pronašao puteve do svoje željene destinacije (najdraže mjesto u planinama, na obali, itd-)

Organizirajmo natjecanje:

<https://www.transum.org/Maths/Investigation/CarPark/>

https://www.transum.org/software/SW/magic_square/magic_square.asp

<https://www.chesskid.com>



ZAKLJUČAK

*Svi svakodnevno koristimo umjetnu inteligenciju.
Umjetna inteligencija nam olakšava život je umjesto nas obavlja različite zadatke.*

Promisli!



Što sada znam?



Što sam naučio/la?



Što još želim znati?



Zaigramo UI bingo!

Što su roboti?
Što roboti imaju zajedničko?
Što oni rade? Kako nam roboti pomažu?
Po čemu se roboti razlikuju od ljudi, a po čemu su slični?

Primjeri:

Cozmo – vizualno prepoznavanje
Alexa/Google home – prepoznavanje govora

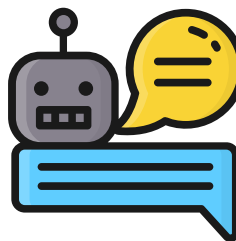
Jesu li roboti jedini oblik umjetne inteligencije?

Što je s društvenim mrežama poput: Instagrama, YouTubea, Facebooka, Googlea, ... ?

Koliko vas ih koristi?

Što imaju zajedničko?

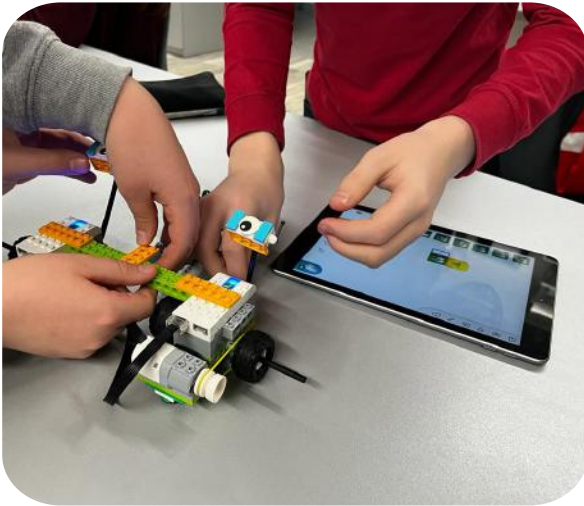
U nastavku ćemo učiti o glavnom zadatku umjetne inteligencije, predviđanju. Kako računala uče?



<http://erasmus-artie.eu>



f



Kratka definicija:

Umjetna inteligencija nastoji predvidjeti nešto u budućnosti ili nešto na što podaci ukazuju.

Primjer nečega što se može predvidjeti (vrijeme, što će se dogoditi u filmu).

Što je skup podataka?

Skup podataka je kolekcija odabranih podataka, čine ga:

- slike
- mjerenja (vrijeme, pogledi, centimetri, itd)
- tekst
- video snimke i sl.

Skup podataka sadrži mnogo različitih dijelova informacija i može biti korišten za uvježbavanje algoritma s ciljem pronalaženja predvidivih uzoraka unutar cijelog skup podataka.

To nisu samo brojevi, podaci o vremenu ili podaci o online videozapisima (pružaju informacije).

Jesi li ikada skupio skup podataka?

Kojih podataka i u koju svrhu?



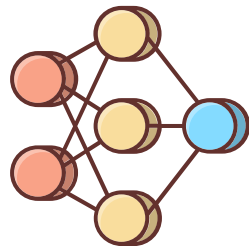


Promislimo o dijeljenju i skupljanju podataka na društvenim mrežama, servisima elektroničke pošte i dr.

Što pokušava prepoznati i za tebe odabrati Instagramov algoritam?

Kakve tipove reklama dobivaš?

Je li to dobro ili loše? Zašto?



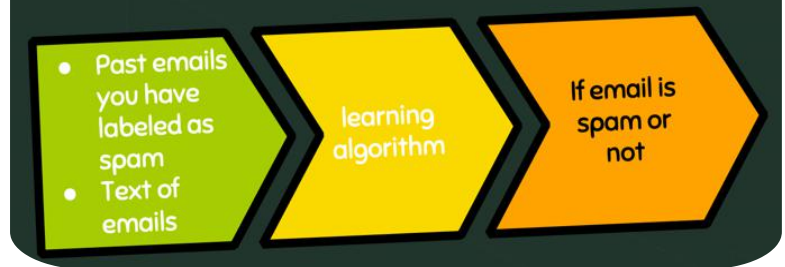
Instagram advertisement



Adding a contact to your phone



Email spam Filter



Zaigramo UI bingo:

Upute:
Pronađi partnera koji ima isti UI sustav na svojoj kartici te zajedno moraju identificirati predviđanja koji će sustav nastojati napraviti te koji će skup podataka sustav za to koristiti.

Polje za igru preuzmite na mrežnoj stranici.





A.I. BINGO

<p>Got a weather forecast from a website or used a weather app</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Sent a voice-to-text message</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Used an online search engine like Google or Bing</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Seen a Google autofill search result</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Had a writing assignment graded by a computer</p> <p>Dataset:</p> <p>Prediction:</p>
<p>Used "safe search" on Google</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Seen a suggested response on Gmail to an email</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Used a Snapchat filter (what's your favorite?)</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Played a motion-sensitive video game e.g. Mario Party, Nintendo, Wii U, etc.</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Had an Emoji suggested instead of a word e.g. "lol" is replaced for an Emoji smiley face</p> <p>Dataset:</p> <p>Prediction:</p>
<p>Seen a sponsored product on Google or Amazon e.g. "since you bought __, we thought you might like..."</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Had an email go to your spam folder (was it actually spam?)</p> <p>Dataset:</p> <p>Prediction:</p>	FREE	<p>Clicked on an Instagram ad (what kinds of ads do you normally see on the app compared to your partner?)</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Seen news articles suggested in a news app (what kinds of articles do you normally see compared to your partner?)</p> <p>Dataset:</p> <p>Prediction:</p>
<p>Had an email labeled as "important"</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Seen a suggested ad on Snapchat (if so, what for? How does this compare to what ads your partner sees?)</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Had a text auto-completed or used autocorrect</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Listened to a recommended song on Spotify (what kind of music do you usually get recommended compared to your partner?)</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Seen a recommended product on Facebook (if so, what for?)</p> <p>Dataset:</p> <p>Prediction:</p>
<p>Seen a "nudge" reminder on Gmail to respond to an email</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Used a fingerprint to unlock a device or opened a device with your face</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Used a map app to find a path to a destination</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Used an app to recognize a song playing</p> <p>Dataset:</p> <p>Prediction:</p>	<p>Communicated with a customer service bot</p> <p>Dataset:</p> <p>Prediction:</p>



ZAKLJUČAK

Ul nastoji predvidjeti ono što će se dogoditi u budućnosti ili nešto na što ukazuju podaci. Skup podataka sadrži mnogo različitih dijelova informacija i može se koristiti za uvježbavanje algoritma sa ciljem pronalaženja predvidivih uzoraka unutar cijelog skup podataka.

Promisli!



Što sada znam?

Što sam naučio/la?

Što još želim znati?



Na koje načine računala uče i rješavaju probleme?

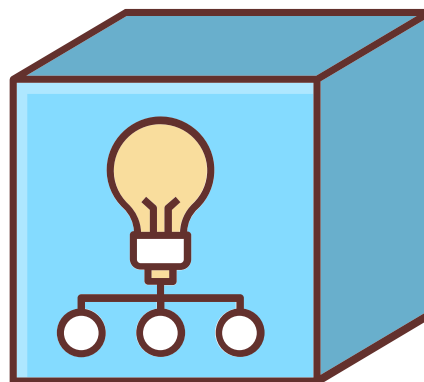
Može li računalo ili drugi pametni uređaj razmišljati ili rješavati probleme?

Može li računalo ili drugi pametni uređaj učiti?

Što je strojno učenje?

Kako računala imitiraju ljudski mozak ili rješavaju probleme?

U nastavku ćemo učiti o strojnom učenju i neuronskim mrežama.





Strojno učenje kao grana umjetne inteligencije

Strojno učenje računalu daje mogućnost učiti bez da je eksplicitno programirano. To je grana računalne znanosti.

Ideja dolazi od rada na umjetnoj inteligenciji. Strojno učenje istražuje kreiranje algoritama koji mogu učiti i predviđati na temelju podataka. Takvi algoritmi prate programirane upute ali mogu predviđati ili donositi odluke na temelju podataka. Oni grade model na temelju unosa uzorka.

Neuronske mreže

Neuronska mreža (ANN ili Artificial Neural Network) je umjetni sustav napravljen od virtualnih apstrakcija stanica neurona. Neuronske mreže se baziraju na ljudskom mozgu i koriste se za rješavanje problema tako da se imitiraju neurone kada se aktiviraju. Za vrijeme računanja mnoge ćelije rade paralelno kako bi proizvele rezultat. Ovo je jedan od mogućih načina po kojem funkcionira UI.

Neuronske mreže mogu samostalno učiti te se po toj mogućnošću razlikuju od normalnih računala. Računala današnjice ne mogu raditi ono za što nisu programirani.

Postoje tri načina prema kojima neuronske mreže uče:

- nadzirano učenje,
- nenadzirano učenje i
- instrumentalno uvjetovanje.

Nadzirano učenje

U nadziranom učenju neuronska mreža se uvježbava korištenjem primjernog ulaza i točnog izlaza. Mreža može shvatiti odnos između ulaza (inputa) i izlaza (outputa). Npr., mreža se može uvježbati tako da joj se pokazuju detalji o kućama i njihova cijena. Nakon što je uvježba može procijeniti cijenu druge kuće tako da analizira informacije poput broja spavaćih soba i stope lokalnog kriminala. Drugi primjer je Autonomous Land Vehicle (ALV). DARPA je financirala ovaj projekt tijekom 1980-ih godina. U demonstraciji 1987. godine, ovo je vozilo proputovala 600 metara brzinom 3km/h preko zahjavnog terena, punog oštrog kamenja, vegetacija i strmog klanca. Ovo vozilo se samo može voziti brzinom od 30km/h. Mreža je promatraja „učitelja“ kako vozi te je cestu promatrala s pomoću laserskog radara. Ovaj se proces se ponovio za različite tipove cesta. ALV je koristio neuralnu mrežu naziva „multi-layer perceptron“ s više slojeva neurona koji su spojeni u serije.



Nenadzirano učenje

Nenadzirano učenje samo uvježbava ulaze (inpute) a mreža mora zaključiti njihov odnos. Ova metoda se koristi za rješavanje grupiranja, procjene i samoorganizirajućih mapa. Na primjer, samoorganizirajuća mapa može biti korištena za kategoriziranje cvijeta irisa po duljini stabljike i boje.

Instrumentalno uvjetovanje

Neuronska mreža koja instrumentalno uči, uči promatranjem radnji. Primjer takvog načina učenja je „mrkva i štap“ učenje (dobro ponašanje se nagrađuje, a loše kažnjava).

Koristeći instrumentalno uvjetovanje, tim istraživača na Sveučilištu u Hertfordshire je nedavno naučio govoriti humanoidnog robota iCuba tako da su koristili „dječji govor“.

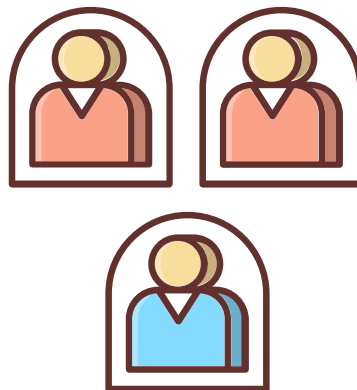
Organizirajmo natjecanje koristeći alate:

- <https://www.autodraw.com>
- <https://quickdraw.withgoogle.com>
- <http://misfire.io>

Istražimo!

Neka učenici odaberu jedan od alata i detaljno istraže njegove mogućnosti

- <https://www.autodraw.com>
- <https://quickdraw.withgoogle.com>
- <http://misfire.io>





Zadatak

1. Napišite kratak opis odabranog alata.
2. Koji bi korisnici mogli biti zainteresirani za ovaj alat?
Na koga bi to moglo najviše utjecati?
Razmislite o najmanje 10 vrsta korisnika.
3. Može li se ovaj alat zloupotrijebiti? Kako?
4. Ako bi se ovaj alat koristio za pomoć drugim ljudima, kome bi mogao pomoći?
5. U slijedećih 50 godina ovaj bi alat mogao učiniti najviše dobra tako što bi...
6. U slijedećih 50 godina alat mogao učiniti najviše štete tako što bi...

Predstavite rezultate učenicima u razredu.
Razgovarajte.
Spremite svoj rad u razredni e-portfolio.



ZAKLJUČAK

*Strojno učenje računalima daje mogućnost da uče.
Neuronske mreže se koriste za rješavanje računalnih problema tako da imitiraju ljudski mozak.*

Promisli!



Što sada znam?



Što sam naučio/la?



Što još želim znati?



Kako algoritmi oblikuju naš svijet?

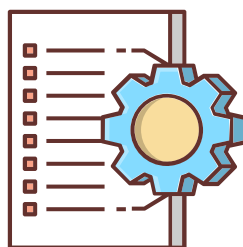
Algoritam je detaljan postupak podijeljen u korake. Izvršavanjem koraka algoritma obavlja se neki određeni zadatak ili se rješava neki određeni problem.

Algoritme možemo stvarati tako da točnim redoslijedom zabilježimo sve korake.

Npr., algoritam rutine jutra djeteta bi mogao ovako izgledati:

- Probuditi se i isključiti alarm
- Odjenuti se
- Oprati zube
- Doručkovati
- Ići u školu

Upoznajmo algoritme temeljene na strojnom učenju.





Što je algoritam?

Algoritam je detaljan postupak podijeljen u korake.

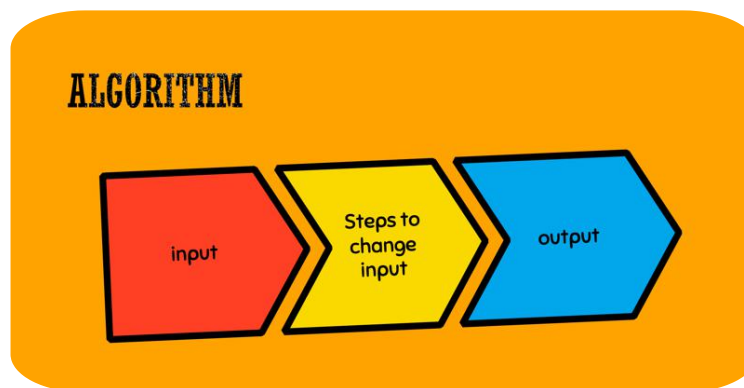
Izvršavanjem koraka algoritma obavlja se neki određeni zadatak ili se rješava neki određeni problem.

Algoritme možemo stvarati tako da točnim redoslijedom zabilježimo sve korake.

Koja su tri dijela algoritma?

Algoritam se sasoji od ulaza, obrade i izlaza.

Algoritmu su potrebni ulazni podaci (input), on zatim prati određene korake ili instrukcije i daje nam željeni izlaz (output).



Ljudi u svakodnevnom životu koriste različite algoritme na sličan način kao i računala.

Algoritmi su iznimno slični receptu za pripremu kolača.

Na primjer, ako bi pekli tortu, algoritam bi uzeo u obzir slijedeće sastojke: brašno, šećer, sol, jaja, itd.

Prvo bi pomiješali suhe sastojke pa ih spojili s mokrim sastojcima poput jaja ili mlijeka.

Zatim bi smjesu izlili u kalup, zagrijali u pećnicu na 180 stupnjeva i u nju stavili kalup sa smjesom.

Naš bi output bila torta!





Zadatak

Napiši svoj „algoritam“ (ili recept) za najbolji sendvič.

Pažljivo specificiraj svoje ulaze (inpute).

Koje inpute (sastojke) trebaš?

Ispiši korake svojeg algoritma.

Koji je izlaz (output) tvog algoritma?

Učenici predstavljaju svoje algoritme. Vode raspravu o sličnosti i razlikama među svojim algoritmima.

Raspravimo

Je li netko od vas u algoritam uključio naredbu pospremanja sastojaka koje ste koristili?

Ako jeste, onda ste vaš algoritam optimizirali za čistoću.

Je li netko izrezao sendvič u neobične oblike ili odrezao koricu?

Ako jeste, onda ste optimizirali algoritam za estetiku ili zaigranost.

Računalni algoritmi se optimiziraju za različite ciljeve, ali to je ponekada teško uočiti.

Što misliš, koji je cilj Googleovog algoritma za pretraživanje?

Interaktivna vježba

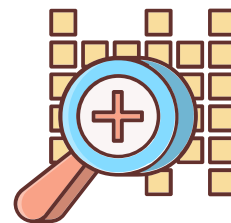
Otvori dva Google pretraživača (jedan gdje su učenici ulogirani u svojim računima i jedan u incognito načinu rada).

Pretraži jednu od slijedećih stvari: vijesti, najbolji film ili pizzeriu.

Što misliš, zašto su rezultati različiti?

Učenici dijele i prezentiraju svoje rezultate.

Raspravljaju po čemu su njihovi rezultati slični, a po čemu različiti.



Teme za raspravu

Googleov algoritam određuju kako se prikazuju web stranice, a važnost im se rangira na temelju vašeg pretraživanja. U manje od sekunde, algoritmi procesuiraju sve vaše informacije koje su međusobno povezane te temeljem toga izbacuje personalizirane rezultate za vaš upit.

Stranice poput Amazona i Netflix-a daju preporuke tako da koriste algoritme za kolaborativno filtriranje koji pregledavaju vaše interese i ukus te temeljem toga donose preporuke za kupnju i program.

Aplikacije za mapiranje poput Google karte trebaju kalkulirati rute kroz gradove, a u obzir uzimaju dužinu puta, promet i prometne nezgode. Alati kao Google Flights u obzir uzimaju rute u mnogo zračnih luka i dodatne informacije poput cijene, vremena i presjedanja.

Raspravimo

Koji uvjeti moraju biti zadovoljeni kako bi netko pobijedio u igri?

Kako možemo pratiti rezultat u našoj igri?

Kako možemo prebrojati koliko se puta pojedino slovo pojavljuje u riječi?

Koje korake moramo poduzeti kako bi zamijenili najmanje i najveće brojeve na listi brojeva?



Interaktivna vježba

Upoznajmo igru ArtBot. U ArtBotu igrači u svim dobnim skupinama uče o umjetnoj inteligenciji.

Zadatak igrača je pronaći ukradene umjetnine. Igrač uvježbava svojeg UI-og pomagača kako bi mogao prepoznati i locirati umjetnine skrivene u labirintu tamnica i time vidjeti kako funkcionira nadzirano učenje i instrumentalno uvjetovanje.

Pokreni igru ArtBot na:

<http://learnml.eu/artbot.php>

Interaktivna vježba

<http://learnml.eu/games.php>

Minecraft Learns ML je kratka igra koja prikazuje kako video igre mogu biti korištene za podučavanje temeljnih principa arhitekture neuralnih mreža koristeći metodu imitacije. U ovoj igri igrači imaju mogućnost da se upoznaju s kreiranjem skup podataka za strojno učenje.

Super Meat Bot je video igra za učenje principa potkrepljenja. U ovoj igri igrači imaju šansu sami dizajnirati razine s nagradama i kaznama kako bi uvježbali umjetnu inteligenciju da sama prevlada sve teže izazove.



ZAKLJUČAK

Od Google pretraživača do jutarnjih rutina, algoritmi su sveprisutni u našoj svakodnevici.

Promisli!



Što sada znam?



Što sam naučio/la?



Što još želim znati?



Kako računala vide svijet oko sebe?

Kako pametni uređaji vide svijet oko sebe?
Kako pametni uređaji donose odluke?
Kako sustavi na bazi strojnog učenja prepoznaju što vide?

Razmisli o pametnim autima, pretraživačima i preporukama za glazbu i video zapise.

Mogu li uređaji učiti?

U nastavku ćemo naučiti kako uvježbati UI modele!

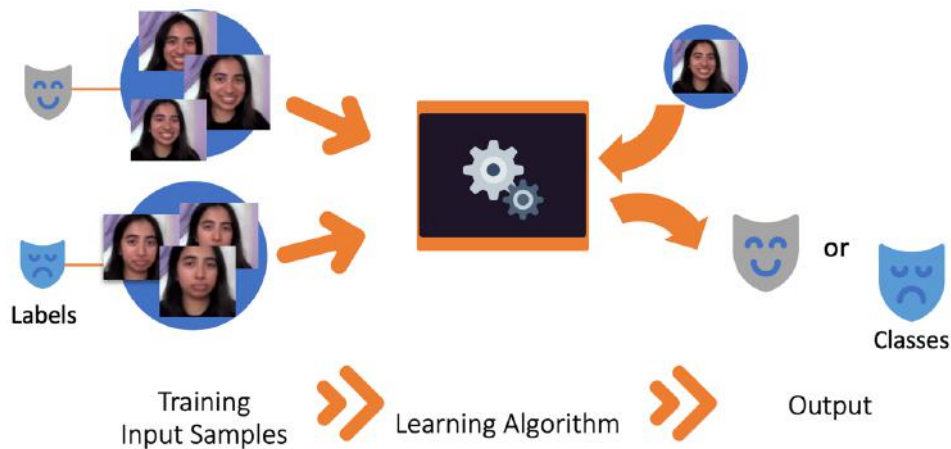




Strojno učenje je tehnika uvježbavanja računalnih sustava da prepoznaju slike i pokret. Kako bi se uvježbao model strojnog učenja, potrebno je pružiti mu mnogo primjera. To je primjerice kada vidiš novu riječ, potrebno ti je puno primjera da je shvatiš.

Zamisli da želimo stvoriti aplikaciju koja te oraspoloži kad si tužan. Kako bi mogao detektirati ako si sretan ili tužan, model strojnog učenja možemo uvježbati fotografija na kojima se smiješ i na kojima se mrštiš. Algoritam bi tako naučio ono što je potrebno. Na kraju svega ovoga, imali bi sustav strojnog učenja koji može prepoznati ako se smiješiš ili mrštiš.

How do machine learning systems recognize what they see?



U ovom su primjeru primjeri inputa (ulazni podaci) slike na kojima se smiješiš ili mrštiš, a na svakoj od slika mora biti oznaka koja označuje ako je u pitanju osmijeh ili mrštenje.

Algoritam uzima te primjere inputa i uči kako predvidjeti ako je dana slika koja prikazuje osmijeh ili mrštenje. To je praktički prepoznavanje uzoraka u nizu primjera.

Kada se algoritam uvježba, možemo ga koristiti za predviđanje i klasificiranje oznaka koje ima slika. Nakon toga u taj sustav možemo ubaciti novu sliku i on će nam reći prikazuje li slika osmijeh ili mrštenje.



Teachable Machine (<https://teachablemachine.withgoogle.com>) je online alat za strojno učenje koji može uvježbati računalo prepoznati različite oblike podataka. Naučiti ćemo kako se koristi Teachable Machine.



Teme za raspravu

Koji podaci mogu biti input za uvježbavanje modela strojnog učenja?

Koje bi bile oznake podataka za uvježbavanje?

Kako računalo zna što vidi?

Razmislimo o koraku u kojem algoritam pronalazi uzorak između označenih inputa.

Na kraju tog procesa možemo razlikovati dva seta slika.

Koje su karakteristike grupa fotografija i kako pomažu računalu prepoznati razliku?

Pomoć: sve će biti vizualne prirode, budući da računalo ne može osjetiti okus ili opipati pahuljice (zadana slika).

Vizualne karakteristike mogu biti veličina, boja, tekstura, oblik i pozadina.

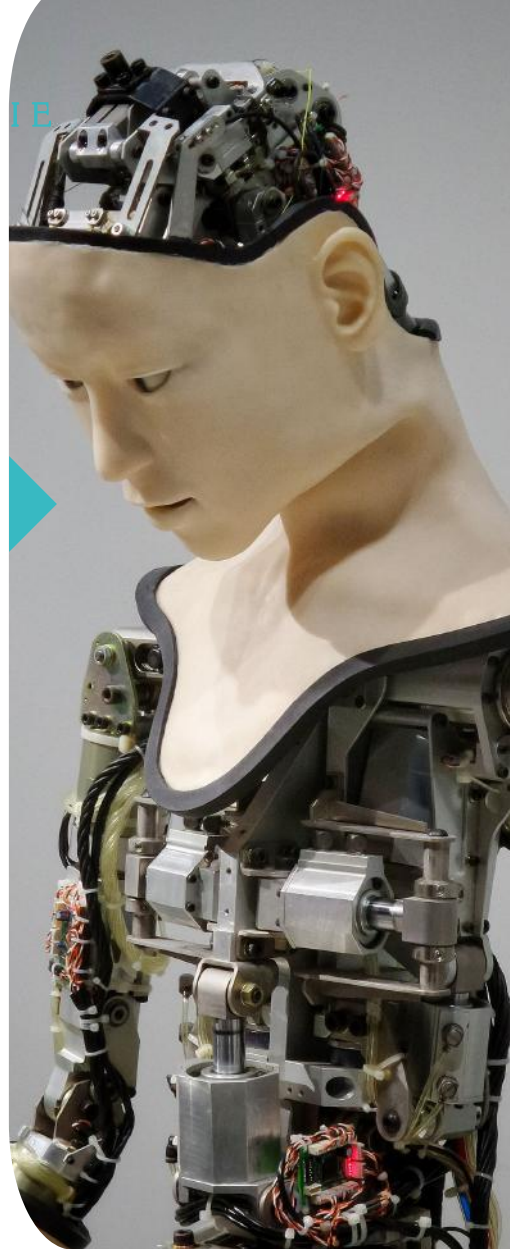
Kako se output sortira u kategorije?

Što nam potpuno utreniran model strojnog učenja može reći o novim slikama?

Možemo ga pitati u koje kategorije svrstava nove slike.

Koje te kategorije mogu biti?

Pomoć: Output često odgovara inputu.





Interaktivna vježba

Uvježbavanje modela za korištenje slika. Idi na:

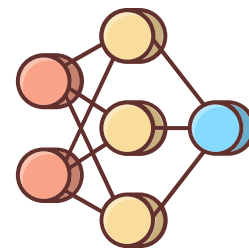
<https://teachablemachine.withgoogle.com>

How do I use it?

- 1 Gather**
Gather and group your examples into classes, or categories, that you want the computer to learn.
[Video: Gather samples](#)
- 2 Train**
Train your model, then instantly test it out to see whether it can correctly classify new examples.
[Video: Train your model](#)
- 3 Export**
Export your model for your projects: sites, apps, and more. You can download your model or host it online for free.
[Video: Export your model](#)

Raspravimo

Možemo li zbuniti model?
 Jest li primijetili da sve slike imaju istu pozadinu?
 Što ako pokušamo promijeniti pozadinu?
 Kako može znati ako se model zbunio?



Get Started – New Project– Image Project



ZAKLJUČAK

Umjetna inteligencija nas uvodi u svijet mogućnosti unutar kojeg možemo uvježbati modele s pomoću različitih tipova podataka, a uvježbavanjem možemo riješiti stvarne probleme.

Promisli!



Što sada znam?

Što sam naučio/la?

Što još želim znati?



Može li računalo prepoznavati poze?

Uvježbali smo modele da prepoznaju slike i zvukove. Možemo li uvježbati modele da prepoznaju poze?

U nastavku ćemo naučiti kako uvježbati naše modele umjetne inteligencije da prepoznaju poze.

Modele koji koriste strojno učenje možemo uvježbati koristeći različite oblike prezentacije podataka. Slike su jedan od oblika prezentacije podataka i slične su fotografijama. Poze, kao što primijetili za vrijeme uvježbavanja podataka, su prezentirane kao plave točkice i linije. To su pojednostavljeni oblici prezentacije podataka poznate kao točka (točka) i rub (linije).

Interaktivna vježba

Uvježbajmo model za klasificiranje različitih pozicija tijela koristeći datoteke ili *striking* poze na web kameri.

Idi na:

<https://teachablemachine.withgoogle.com>

Prezentiraj svoj model suučenicima. Raspravite.

Pohrani svoj rad na razredni e-portfolio.



Teme za raspravu

Što misliš da je drugačije s modelom za poze?

Misliš li da bi se model zbunio ako bi koristili drugačiju pozadinu?

Umjetna inteligencija nam otvara vrata u svijet mogućnosti: mi uvježbavamo modele učiti tako da procesuiraju različite tipove podataka i te modele primjenjujemo pri rješavanju stvarnih problema.

Imali li nedostataka kod modela strojnog učenja?

Koje su negativne strane modela koji uče strojnim učenjem?

Što može poći u krivu?

ZAKLJUČAK

Modele slika smo uvježbali tako da smo im za to dali fotografije, dok se modeli poza uvježbavaju s pomoći plavih točaka i linija (koje prepoznaju kao poze).



Promisli!



Što sada znam?



Što sam naučio/la?



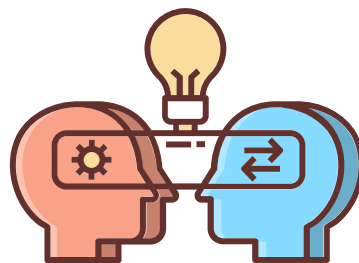
Što još želim znati?



Kako etično koristi podatke za uvježbavanje modela?

Što je privatnost?
Kako štitimo našu privatnost?
Može li UI potkopati našu privatnost?

Razgovarati ćemo o etičkoj strani korištenja podataka koje koristimo za uvježbavanje modela.





Umjetna inteligencija je ljudski proizvod i kreirana je sa ciljem koji je orijentiran prema ljudima. Siri, Alexa i Google Home nisu samo roboti, osoba sa širokim spektrom znanja iz računalnog inženjerstva je zaslužna za inteligentne glasove koji izlaze iz ovih elegantnih uređaja. Budući da su ove strojeve programirali ljudi, oni su skloni greškama koje mi radimo i imaju određene pristranosti.

Kod rada s umjetnom inteligencijom je najbolje raditi s čim većim brojem podataka. Kako bi dizajnirali umjetnu inteligenciju, ljudima je potrebna velika količina podataka, s pomoću koje postižu maksimalnu preciznost. Koji je izvor tih podataka? Vi i ja, naši osobni podaci, naše rutine, naši interesi i njihova online manifestacija. S tim dolaze problemi u privatnosti i moralna dilema; koliko je etično koristiti podatke koji nisu dani nužno za razvijanje dane tehnologije. Budući da djeca sve ranije i ranije dobivaju mobitele, važno je da znaju pojedinost tehnologija koje ih okružuju.

Podaci mogu biti važan alat za računalne sustav, ali moraju se koristiti odgovorno i etički. Privatnost je važan element u korištenju podataka zato nitko ne želi da se javno dijele njihovi osobni podaci.



Teme za raspravu

Koliko vas bi prihvatilo da hakeri imaju pristup vašim korisničkim imenima? Tvojem broju telefona? Tvojem korisničkom imenu na TikToku? Fotografiju tvog lica? Što ako bi ljudi pristup njim, ali i mogli pratiti što objavljuješ i što radiš koristeći te podatke?

Interaktivna vježba

Stvorimo zajednički online poster sa savjetima za zaštitu privatnosti. Predstavljaju radove i raspravljaju.

ZAKLJUČAK

Uređaji na bazi umjetne inteligencije nisu sveznajući roboti. To su uređaji koje su izradili ljudi i stoga, kao ljudi oni mogu griješiti. Još važnija je činjenica da oni uče na svojim greškama i s pomoći njih se unaprijeđuju.

Promisli!



Što sada znam?



Što sam naučio/la?



Što još želim znati?



Stvorimo vlastiti UI projekt pomoću Scratch-a!

Interakcija između ljudi i umjetne inteligencije je dvosmjerna: mi reagiramo na učinke i ponašanja UI sustava u robotima, a računala i UI reagiraju na razne oblike ljudskog ponašanja.

Za što mi ljudi koristimo pokret u svakodnevnom životu? Krećemo se kako bi stigli nekamo! Bilo to hodajući, vozeći auto ili bicikl, plivajući, koturajući se ili nekako drugačije, mi ulažemo trud kako bi se dopremili s jednog mjesta na drugo.

Također se prilično krećemo kada se bavimo sportom. Ne zaboravimo ulogu kretanja i micanja u umjetnosti! Promislimo o načinima kretanja kada slikamo, radimo glazbu ili glumimo.

Kreativni procesi imaju razne oblike: neki uključuju pokret, dok drugi ne. Umjetnički činovi koji koriste kretanje su prilično moćni; zamislite viseću instalaciju koja se njiše na vjetru ili usklađenost orkestra ili baletnu grupu kako se besprijekorno kreće uz melodije opere.

Ponekad se izražavamo vizualno bez da uopće razmišljamo o tome. Također često koristimo izraze lica kao znakove koji otkrivaju kako se osjećamo. Drugi ljudi mogu pročitati te znakove i razumjeti naše emocije ili namjere.



ARTIE



Kretnja je vrlo moćan alat u komunikaciji i način na koji ljudi šalju signale jedni drugima. Pomažu prenositi poruke i nose ideje od jednog do drugog čovjeka ili čak na stotine njih. Stoga, pokreti su ključan medij putem kojeg ljudi međusobno komuniciraju i razumijevaju se.

Čini se da se entiteti na bazi umjetne inteligencije mogu kretati ako su programirani za to. No može li UI razumjeti ljudske pokrete? Što misliš?



UI ne „razumije“ pokret na način kako ga mi razumijemo, no može se uvježbati da izdvoji određene uzorke kretanja i reagira na njih. Slike i poze koje smo koristili na Teachable Machines su znakovi koje računalo koristi kako bi izvuklo značenje iz pokreta, one predstavljaju vizualne komponente pokreta ili poziciju tvoga tijela tijekom kretanje, no zapravo ne kodiraju sam pokret.

Proučit ćemo UI sustave koji razumiju ljudski pokret, izraze lica i druge interakcije temeljene na pokretu.

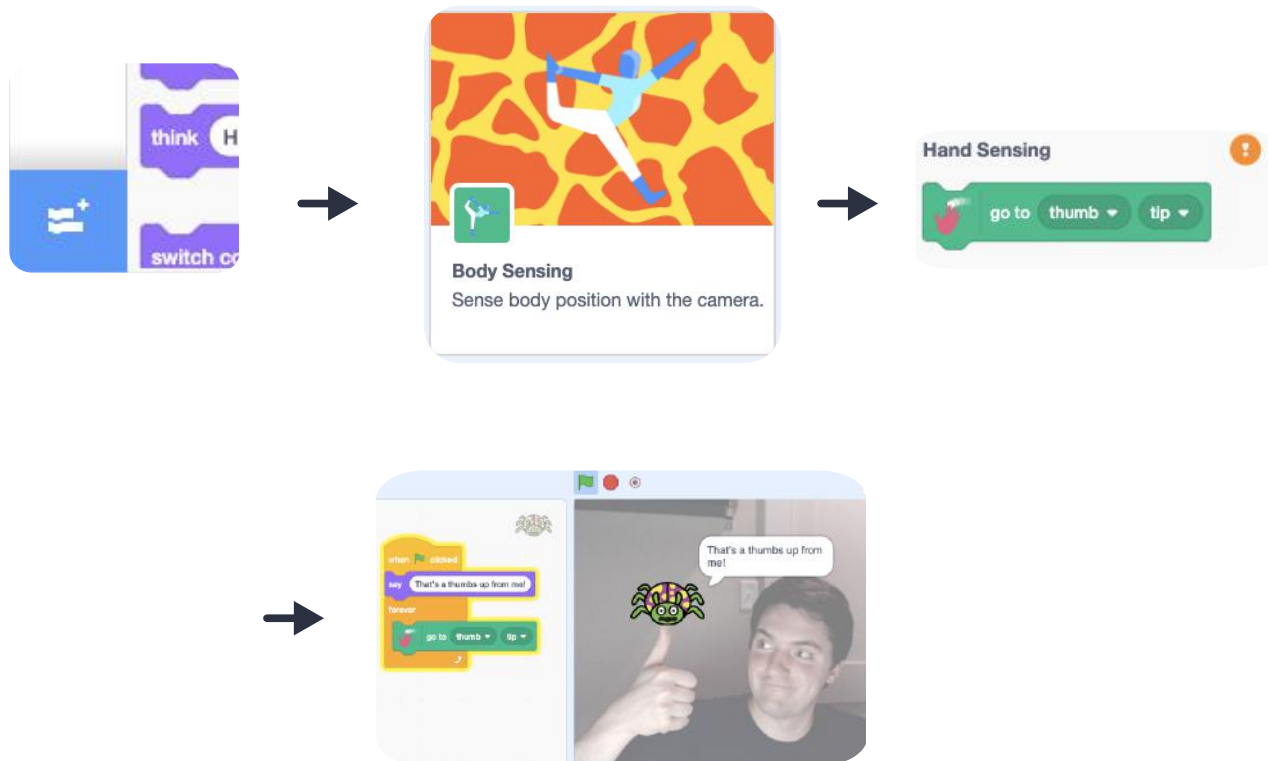
Scratch je alat koji se koristi za stvaranje. Možemo iskoristiti Scratch kako bi stvorili priče, igre, animacije, umjetnička djela, glazbu i interaktivne UI sustave. Koristit ćemo posebnu verziju Scratch-a koji nadodaje posebne UI značajke kao dodatke.

Učili smo o razlici između Teachable Machine modela koji se uvježbavaju s pomoću podataka iz slike i poze. Slikovni modeli uvježbani su putem cijele fotografije, dok pozni modeli gledaju samo točke i linije koje prepoznaju kao pozu tijela.

Kako bi preveli slike u poze, Teachable Machine koristi unaprijed uvježban model zvan „PoseNet“. To je model strojnog učenja koji je uvježban na vrlo mnogo primjera slika i njima odgovarajućim kosturima poza.

PoseNet uzima fotografiju i pretvara ju u plave točke i linije. Te točke nazivaju se „ključne točke“. Iako smo koristili te ključne točke kao unos za naše Teachable Machine modele, možemo koristiti iste i kao dijelove Scratch projekata!

Kako bi koristili PoseNet u našim projektima moramo samo uvesti neke dodatne blokove! Kliknemo na „Add Blocks“ plavi gumb na dnu ekrana. Zatim odaberemo „Body Sensing“ ekstenziju.





Kada to dodamo, primijetimo da nam se kamera uključila s desne strane ekrana. Zelena kvačica pokraj „Body Pose Sensing“ znači da PoseNet model ima predviđanje gdje su ključne točke na fotografiji! Ako ima narančasti upitnik, to bi moglo značiti da postoji problem s kamerom ili aplikacijom ili pak da model nema dovoljno pouzdano predviđanje. Ako pokušaš prekriti kameru rukom, na primjer, titrati će između kvačice i žute oznake.

Ako zaustavimo projekt i kliknemo na taj blok, naš će se lik pomaknuti tamo gdje misli da vidi ključnu točku lijevog ramena u sceni kamere.

Ako postavimo taj blok u beskonačnu petlju kao što smo učinili prije, vidjet ćemo da nam prati rame.

Blokovi za ruke koriste drugi pozni model koji je sličan PoseNetu iz Google MediaPipe tima.

Taj model nije uvježban na fotografijama tijela, već na fotografijama ruku. Crvene točke u ovim primjerima su također ključne točke, upravo kao ključne točke koje dobivamo iz PoseNet modela. Budući da svaki model koji dodaš znači da tvoje računalo mora pokrenuti svaki model više puta u sekundi (tzv. okvirna stopa modela), poželjno je sačuvati napredak i osvježiti stranicu preglednika prije učitavanja modela tijela.

Nakon uvažanja modela za opažanje ruku, primijetit ćete da imamo novi blok:

Valja uzeti u obzir da će indikator biti narančast ako ruke nisu vidljive na kameri, budući da pokazuje zeleno jedino kada predviđa da postoji ruka u kameri.

Interaktivna vježba

- Otvori Scratch.
- Otvori Novi projekt
- Dodaj „Body Sensing“ blokove
- Napravi svoj projekt
- Testiraj i spremi svoj projekt
- Prezentiraj svoj projekt drugim učenicima. Prodiskutirajte. Međusobno se ocijenite.
- Spremi svoj uradak na razredni e-portfolio

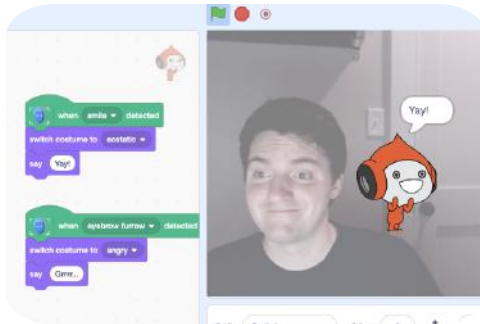
Face (Lice) model je model kompanije Affectiva koja se bavi emocionalnom UI. Slično kao PoseNet i Hand Model MediaPipe-a, Affectivin model ti daje ključne točke na licu (npr. obrva, usta, nos, itd.), no ide i korak dalje pa nam daje „izraze“, kao npr. „otvorena usta“, „podignute obrve“, „treptanje“, itd.

Možemo koristiti ključne točke lica isto kao što smo koristili podatke ruke i poze tijela. Možemo koristiti izraze lica kao što je „otvorena usta“ kao događaj (događaj) u našim Scratch projektima, upravo na način kako blok „when Green Flag“ („kada Zelena Zastava“) čini da nam kod radi. Probajmo promijeniti kostime kada se smijemo i skupimo obrve.

Budući da je ovaj model složeniji i da su neke klase unosa subjektivnije, ne možemo očekivati od onog višeg nivoa saznanja oko izraza lica (i emocija osobito) da bude toliko pouzdan koliko su pouzdane ključne točke u modelima.



Osmislimo, izradimo i predstavimo svoje radove



i



Interaktivna vježba

- Otvori Scratch
- Otvori Novi projekt
- Dodaj „Face Sensing“ blokove
- Napravi svoj projekt
- Testiraj i spremi svoj projekt
- Prezentiraj svoj projekt drugim učenicima u razredu.
- Raspravi s učenicima u razredu.
- Međusobno vrednujte svoje radove.
- Spremi svoj uradak na razredni e-portfolio

ZAKLJUČAK

Možemo izraditi vlastite programe koji će tražiti određena svojstva, primjerice položaj očiju i smjer u kojem pomičemo prst. Mi određujemo kako se računalo ponaša s obzirom na ljudsku kretanju, što znači da mi kontroliramo što UI sustav radi.

Promisli!



Što sada znam?

Što sam naučio/la?

Što još želim znati?

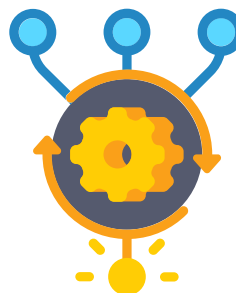


Zabavimo se uz umjetnu inteligenciju!

Ponovimo naredbe i vještine rada u Scratchu:

- dodavanje blokova (*Body Sensing*, *Face Sensing*, *Hand Sensing*), otvaranje novog projekta,
- dodavanje BACKDROPOVA (pozadine) i
- likova (*sprites*),
- dijeljenje projekta i
- preuzimanje projekta na računalo.

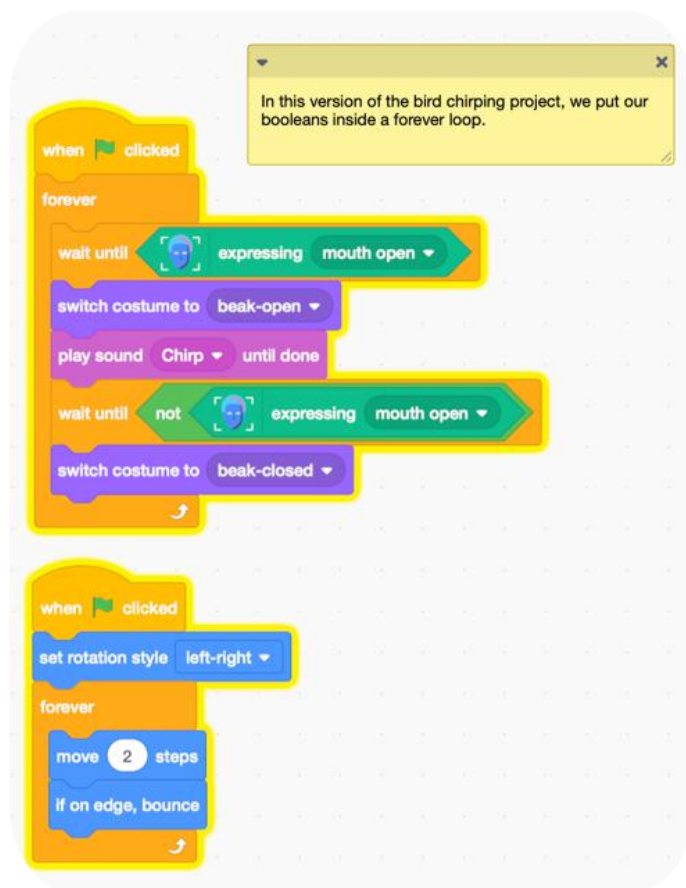
Izradimo zabavnu igru koristeći „Face Sensing“ blokove, petlje i Booleove operatore.





Interaktivna vježba

- Otvori Scratch.
- Stvori novi projekt.
- Dodaj *Sprite* i *Backdrop*.
- Izradi svoj projekt koristeći blokove: *Motion*, *Looks*, *Sound*, *događaj*, *Control*, *Body / Hand / Face Sensing*).
- Testiraj i pohrani svoj projekt.
- Prezentiraj svoj projekt učenicima u razredu. Raspravite. Međusobno vrednujte vaše radove.
- Pohrani svoj rad na e-portfolio razreda.



ZAKLJUČAK

Možemo stvarati programe koji će tražiti pojedinosti, primjerice izraze lica. Mi stvaramo program i mi odlučujemo na koji će način računalo reagirati na određeni pokret. Prema tome možemo zaključiti da ljudi kontroliraju sustave umjetne inteligencije.

Promisli!



Što sada znam?

Što sam naučio/la?

Što još želim znati?



Reciklirajmo zajedno !

Ponovimo naredbe i vještine rada u Scratchu i kako se uvježbavaju modeli s pomoću Teachable Machine ekstenzije.

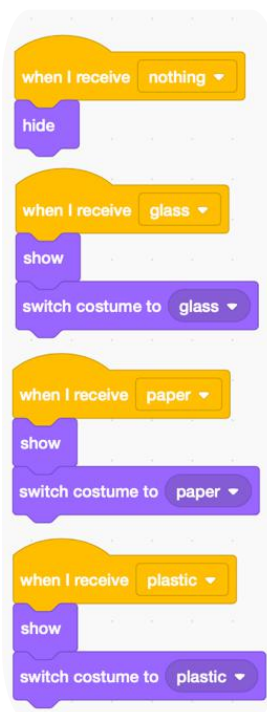
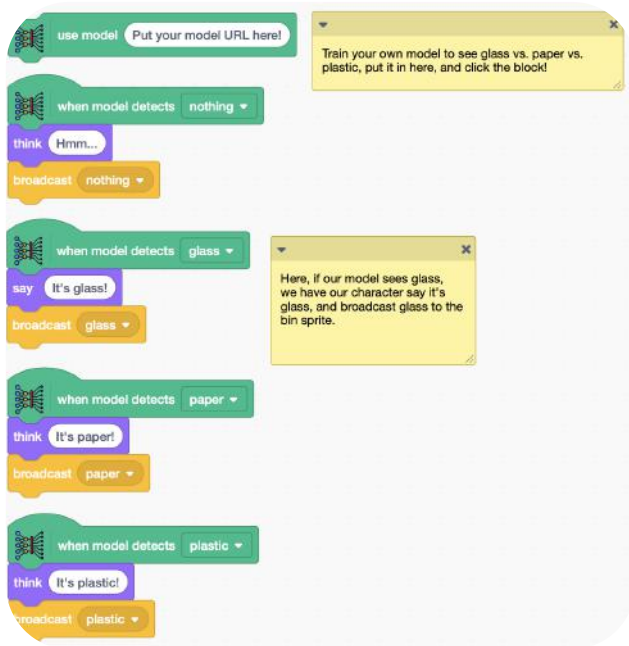
Pažljivo upravljanje otpadom je iznimno važna vještina koju bi trebao imati svaki stanovnik našeg planeta. U nastavku ćemo koristiti Scratch i Teachable Machine kako bi kreirali vlastitog pomoćnika za recikliranje.





Praktičan rad:

- Idi na: <https://teachablemachine.withgoogle.com>
- Uvježbaj model da prepozna staklo papir i plastiku s pomoću web kamere
- Napravi novi projekt u Scratchu
- Dodaj *Video Sensing* ekstenziju
- Dodaj SPRITE/LIK (*assistant*) i povezane blokove
- Kopiraj i zalijepi URL svojeg modela
- Dodaj novi sprite i povezane blokove



ZAKLJUČAK

Koristeći Scratch i Teachable Machine možemo stvoriti i koristiti vlastitog pomoćnika za recikliranje.

Promisli!



Što sada znam?

Što sam naučio/la?

Što još želim znati?



Čavrljajmo!

Izradimo svog chatbota koji odgovara na pitanja iz zadane teme

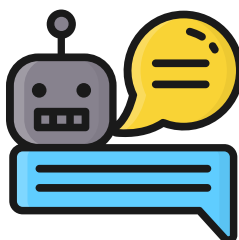
Možemo li razgovarati s našim računalima? Kako?
Koristiš li chatbotove? Kako i kada?

Isprobaj chatbot:

<http://talktothetrex.com> je primjer dobrog bota. Razmislite, kako bi mogli napraviti svoj chatbot?

Chatbot je software koji simulira ljudske razgovore s korisnicima koristeći tekstualne poruke u chatu. Najvažniji im je zadatak odgovoriti na postavljena pitanja.

U nastavku ćemo kreirati chatbot koristeći Scratch i Teachable Machine.



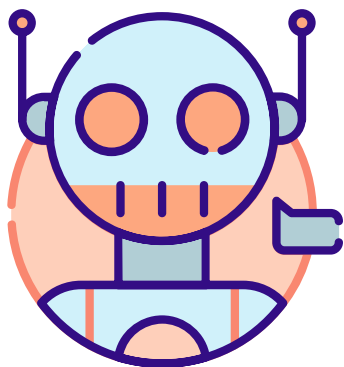


Teme za raspravu:

Odluči o temi za svojeg chatbota. Odaberi nešto što dobro znaš dovoljno dobro da bi odgovarao na pitanja postavljena na temu. Može biti mjesto, životinja, povijest, programski jezik, itd.

Razmisli koja pitanja bi mogla biti postavljena za tvoju temu. Primjerice, pogledaj pitanja koja se tiču sova:

- Što sove jedu?
- Gdje sove žive?
- Koliko dugo sove žive?
- Kakve sve sove postoje?
- Koliko sove mogu narasti?



Upute (koraci):

Idi na <https://machinelearningforkids.co.uk/> u pretraživaču.

Klikni na „Get started”.

Klikni na „Try it now”.

Klikni na „Projects” na vrhu izborne trake.

Klikni na „+ Add a new project” gumb.

Imenuj svoj projekt i postavi ga tako da uči prepoznati tekst. Klikni „Create” gumb.

Klikni na svoj novi projekt na listi projekata.

Klikni na „Train”.

Klikni na „+ Add new label”.

Utipkaj riječ koja sumira prvu stvar na tvojoj listi iz drugog koraka zatim klikni na „Add”. Koristili smo “hrana” kako bi sumirali pitanje poput “Što sove jedu?”

Ponovi ovaj proces za sva pitanja na drugom koraku.

Nije važno koje ćeš riječi izabrati, važno je da ih razumiješ.

Klikni na „+ Add example”.



- Utipkaj primjer pitanja koji bi se mogao postaviti na temu.
- Klikni na „Add” u bucket.
- Ponovi radnju tako da imaš pet primjera pitanja.
- Ponovi radnju tako da imaš pet primjera in every bucket.
- Kliknu na link „< Back to project”.
- Klikni na „Learn & Test”.
- Klikni na „Train new machine learning model”. Ako imaš dovoljno primjera, računalo će početi kako prepoznati pitanja s pomoću danih primjera.
- Pričekaj da uvježbavanje završi. To može potrajati nekoliko minuta. Biti će dovršeno kada na statusnoj traci piše „Available”.
- Klikni na link „< Back to project”.
- Klikni na „Make”.
- Klikni na „Scratch 3”.
- Klikni na „Open in Scratch” gumb.
- Otvori „Owls project” predložak. Klikni na „Project templates“ i pronađi „Owls“ na listi predložaka.
- Odaberi ovaj izrezak skripte, ali još nemoj ništa prilagati. Pazi da odabereš „owl says” za narančasti blok.
- Dupliciraj to četiri puta i spoji ih zajedno. Na to klikni desnom tipkom miša i odaberi „Duplicate”.
- Nadopuni svaku kopiju bloka. Povuci oznaku za jedno od svojih pitanju u gornji prostor i utipkaj odgovor u donji dio prostora.
- Povuci ovaj novi blok u zeleni blok koji je pripremljen za ovo. Obriši „Sorry. I haven't been taught anything yet.” blok i zamijeni ga svojim novim komadom skripte.
- Nacrtaj svoj chatbot. Nacrtaj neki lik. Ako svojem liku daš kostime, možeš ga animirati kada priča.
- Testiraj svoj chatbot! Klikni na zelenu zastavu i postavi mu pitanje.
- Prezentiraj svoj projekt učenicima u razredu. Raspravite. Međusobno se evaluirajte.
- Pohrani svoj rad na e-portfolio razreda

f

<http://erasmus-artie.eu>

ZAKLJUČAK

Chatbot je software koji simulira ljudske razgovore s korisnicima koristeći tekstualne poruke u chatu. Najvažniji im je zadatak odgovoriti na postavljena pitanja. Možemo napraviti chatbot koristeći Scratch i Teachable Machine.

Promisli!



Što sada znam?



Što sam naučio/la?



Što još želim znati?



Pametni dom

Koristimo li pametne uređaje svaki dan?
Što su pametni uređaji?
Zašto ove uređaje zovemo pametnima?
Mogu li kuće biti pametne?
Koje su kuće "pametne"?

Pametan dom se odnosi na pogodan način postavljanja aparata i uređaja u domu koji se mogu kontrolirati automatski ili iz udaljenosti koristeći internetsku konekciju, mobitele ili druge umrežene uređaje.

U nastavku ćemo kreirati pametne uređaje kao dio pametnog doma koristeći Scratch.





Upute (koraci):

Sušilo za kosu:

```

when clicked
  forever
    ask Enter your command and wait
    if answer = Turn on the fan then
      broadcast turn-fan-on
    if answer = Turn off the fan then
      broadcast turn-fan-off
    if answer = Turn on the lamp then
      broadcast turn-lamp-on
    if answer = Turn off the lamp then
      broadcast turn-lamp-off
  
```

Svjetiljka:

```

when clicked
  switch costume to lamp-off
when I receive turn-lamp-off
  switch costume to lamp-off
when I receive turn-lamp-on
  switch costume to lamp-on
  
```

Ventilator:

```

when I receive turn-fan-on
  set on to yes
  repeat until on = no
  next costume
when I receive turn-fan-off
  set on to no
when clicked
  set on to no
  
```

Prezentiraj svoj projekt učenicima u razredu. Raspravite.

Međusobno vrednujte svoje radove.

Pohrani svoj rad na e-portfolio razreda.



ZAKLJUČAK

Pametna kuća odnosi se na prikladnu kućnu postavu u kojoj mogu biti aparati i uređaji kojima se automatski upravlja daljinski s bilo kojeg mjesta s internetskom vezom pomoću mobilnog telefona ili drugog umreženog uređaja.

Promisli!



Što sada znam?

Što sam naučio/la?

Što još želim znati?



Upoznajmo Cody Rocky-a

Možeš li se sjetiti nekih robota osim Codey Rockyja?
Za što se koriste ti roboti?
Kako ti roboti mogu razumjeti naše upute?

U nastavku ćemo upoznati i naučiti koncept programa i što oni mogu raditi, kao i osnove programa Codey Rocky i mBlock 5.





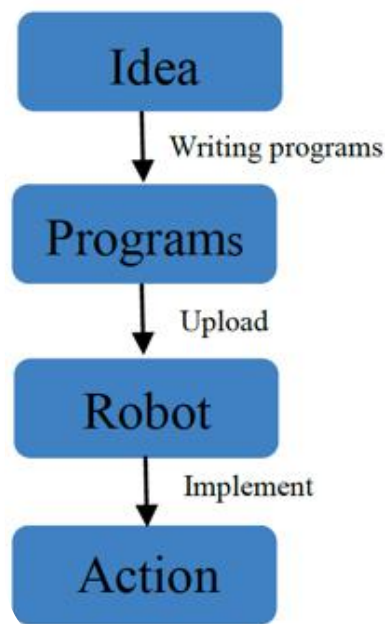
Program je umjetni jezik koji koristimo kako bi rekli robotima što da rade. Upute koje je potrebno izvršiti prikazujemo kao naredbe koje su dio programa.

Zatim učitaemo program u robota i omogućavamo mu da radi svakojuke stvari koje smo isprogramirali.

Znate li sada što je odgovor?

Koja je tajna Codey Rockyja?

Želite li napisati kod i učitati ga u Codey Rocky?



Codey Rocky je edukativni programibilan robot. Možete koristiti softver kako bi programirali robot, upravljajući njime tako da radi razne stvari. Također je i dobar prijatelj koji može pomoći djeci da nauče programirati. Uz mBlock 5, djeca mogu savladati osnove programiranja i razviti logičko razmišljanje kao i računalno razmišljanje. Osim toga, Codey Rocky podržava tehnologije kao što su UI i IoT koji izlažu djecu najnovijim tehnologijama.



Codey: Kao mozak robota, Codey je opremljen raznim sensorima i programibilnim blokovima. Može raditi samostalno ili zajedno s Rockyjem kako bi mogao uraditi više zadataka. Sada uzmite svoje Codeyje. Pogledajmo kakve senzore ima.



Feature	Block Name	Applications and Core Functions
Input	IR transmitter and IR receiver	Facilitates communication between Codeys and remote controls
	Gear potentiometer	Adjusts the input signals
Input	Gyroscope	Detects how Codey moves and the angles.
	Buttons	Buttons can be programmed to control the facial expressions, motions, and sounds of Codey Rocky.
	Light Sensor	The sensor is used to measure the volume of sounds in the surrounding environments.
	Sound Sensor	The sensor is used to measure the light intensity of surrounding environments.
Output	Speaker	The sensor can be programmed to play music.
	RGB indicator	The indicator can glow in different colors.
	LED matrix screen	The images, texts and time displayed on the screen are all customizable.

Rocky služi kao postolja Codeyja. Daje dodatne mogućnosti Codeyju kao što su izbjegavanje prepreka, prepoznavanje boja, praćenje linija i dr.



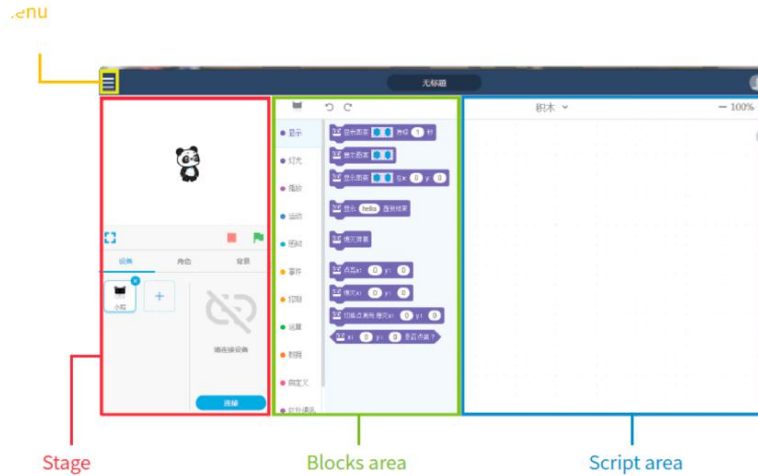
Features	Name	Applications
Input	IR Color Sensor	The IR Color Sensor integrates a color sensor, a grayscale sensor, an IR proximity sensor. By toggling the IR Color Sensor, you can make Codey Rocky perform a variety of fun tasks, like avoiding obstacles, following lines and more.
Input	Motor	The motor is used to control the motion of Rocky.

O mBlock 5

mBlock 5 je alat za programiranje koji podržava Python programske jezike koji su temeljeni na blokovima. Razvijen je na temelju Scratch 3.0, open-source alata za softver kojeg su zajedno razvili MIT i Google. Koristeći mBlock 5, možete napisati programe koji govore Codey Rockyju ili drugim robotima da rade što god vi željeli. Čak možete i iskoristiti softver kako bi napravili priče, igre i animacije koje su zabavne i jedinstvene. Nadalje, mBlock 5 izlaže djecu tehnologiji poput UI, dubokog učenja uvježbavanja modela. Ukratko, mBlock 5 je jedna od najboljih opcija za programere početnike. Neka učenici otvore mBlock 5 PC i objasnite im sučelje.



Sučelje za pokretanje:



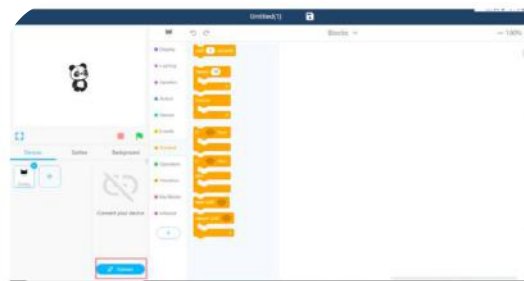
1. Stage (pozornica): U ovom području možete prikazivati projekte, spajati uređaje, učitavati programe, dodavati likove te pozadine
2. Blocks area : Ovdje možete pronaći potrebne blokove po boji ili kategoriji
3. Script area: U ovaj dio povlačite blokove iz programa
4. Device/Sprites/Backgrounds Setting area (područje za postavljanje uređaja, likova i pozadina): Ovdje možete pronaći potrebne uređaje, likove i pozadine

Zadatak

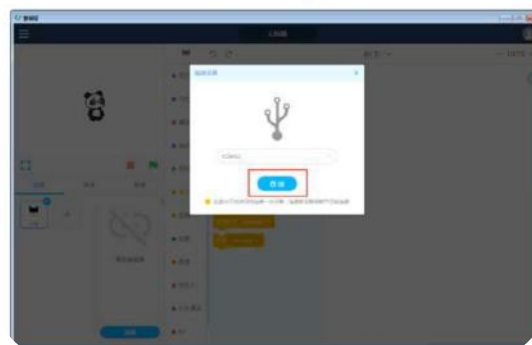


1. Spojite se na računalo: Spojite Codeyja na računalo preko USB kabela. Zatim uključite Codeyja.
2. Odaberite serijski port: Otvorite mBlock 5, pritisnite Connect i odaberite ispravan serijski port.

<http://erasmus-artie.eu>

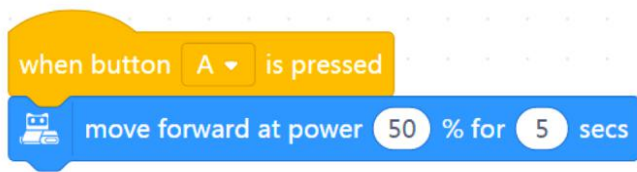


Click Connect

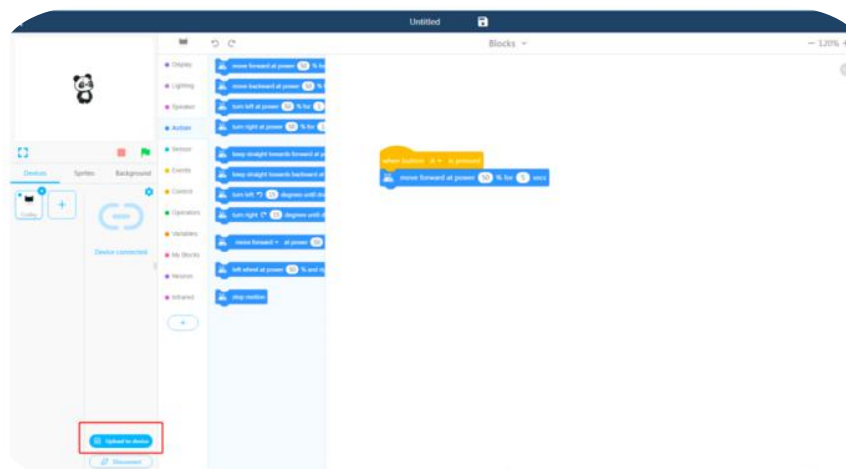




3. Iskoristite mBlock 5 kako bi napravili dio programa kao ispod:



4. Učitajte program na Codey



5. Izvadite USB kabel i postavite Codey Rocky na stol. Pritisnite tipku A i opazite kako Codey Rocky reagira. Neka učenici riješe gornje zadatke tako da napišu programe, radeći u paru.

O događajima

Na primjer: Kada je mračno i uđemo u sobu, moramo uključiti svjetlo. Kako bi upalili svjetlo, moramo pritisnuti gumb za svjetlo. U ovom slučaju, pritisak gumba je događaj i to da je svjetlo upaljeno je rezultat.

Igrajmo društvenu igru s Codey Rockyjem

Pravila igre:

Organizirajte se u 3 ili 4 grupe.

Nacrtajte neke oblike na ploči, primjerice trokut, krug, kvadrat i zvijezda.

Definirajte 3 ili 4 događaja:

Kad stavite ruku na trokut;

Kad stavite ruku na krug;

Kad stavite ruku na kvadrat;

Kad stavite ruku na zvijezdu;

Ta četiri događaj će aktivirati slijedeće akcije:

Kad stavite ruku na trokut —druga grupa učenika ustane;

Kad stavite ruku na krug —četvrta grupa učenika ustane;

Kad stavite ruku na kvadrat —prva grupa učenika ustane;

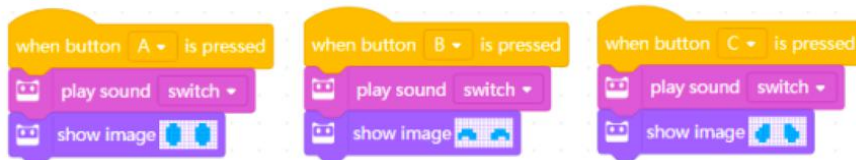
Kad stavite ruku na zvijezdu —treća grupa učenika ustane;

**Zadatak 1:**

Naučite postavljati događaj. Ispišite programe kako bi učinili da Codey Rocky mijenja izraze lica na temelju evenata. (kada je gumb A/B/C pritisnut).

**Zadatak 2:**

Ispišite programe kako bi učinili da Codey Rocky reagira na događaj (kada je gumb A/B/C pritisnut). Na primjer da mijenja izraz lica ili radi različite zvukove. Neka si učenici međusobno pokažu projekte.

Primjeri:

O čemu je tvoj projekt?

Jesi li naišao/naišla koji problem?

Kako si ih riješio/la?

**ZAKLJUČAK**

Program je umjetni jezik koji koristimo kako bi rekli robotima što da rade. Prevodimo naše upute u dio programa. Zatim učitamo program u robota i omogućavamo mu da radi svakojake stvari koje smo isprogramirali.

Sučelje mBlock 5 sastoji se od: Stage area, Blocks area, Scripts area, Device/Sprites/Backgrounds Setting area.

Događaj je početak dijela programa. Kad pišete programe, prvo morate odabrati događaj.

Promisli!**Što sada znam?****Što sam naučio/la?****Što još želim znati?**



Codey Rocky upoznaje slijed naredbi i petlje

Što je „događaj“?

Možeš li se prisjetiti nekih događaj/eventa u svakodnevnom životu?

Koje smo događaj koristi u prošloj lekciji?

Jednostavni odgovori:

Događaj je radnja koja može započeti stvari.

Pritiskanjem gumba možemo upaliti žarulju.

U ovom je slučaju pritiskivanje gumba događaj, dok je upaljena žarulja rezultat.

Događaj korišteni u posljednjem nastavnom satu uključuju: pokretanje programa, pritiskanje A/B/C gumba.

Koji su koraci stavljanja mlijeka u frižider?

Objasnite učenicima:

Kako bi stavili mlijeko u frižider morate poduzeti slijedeće korake: otvori frižider, stavi mlijeko unutra i zatvori frižider.

Možeš li se sjetiti neke druge radnje koja ima određeni redoslijed?

U nastavku ćemo upoznati konceptom sekvence (slijeda naredbi) i petlje te razviti program koji ih koristi.





Što su sekvence (slijed naredbi) i petlje?

Kako nastaju animacije?

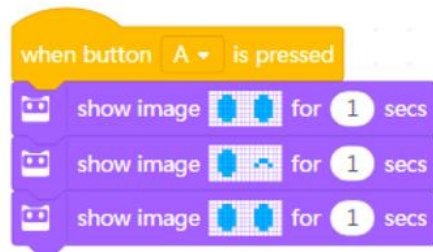
U tradicionalnoj animaciji su animatori animirali prateći ove korake:

Prvo su stavili statični prizor kao pozadinu i preko toga su stavili novi papir.

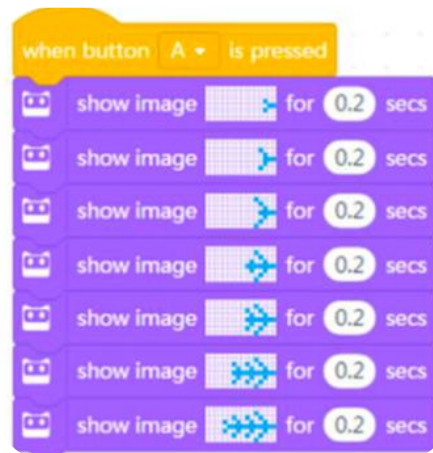
Animatori bi zatim napravili skoro identičan obris, crtež se minimalno mijenjao. Zatim bi napravili isti korak sve dok ne dovrše niz slika koje se minimalno razlikuju. Zatim bi iznimno brzo prelistali papire i time dobili animaciju. Koristeći isti princip koristimo blok „show image () for () secs” u našem programu kako bi animirali animacije.

Prvi zadatak: Namigivanje

Kreiraj animaciju koristeći blok „show image () for () secs”. Metoda je jednostavna: koristi sliku bloka kao bazu, dupliciraj blok, malo promijeni sliku. Ponovi korake i posloži blokove u sekvencu.

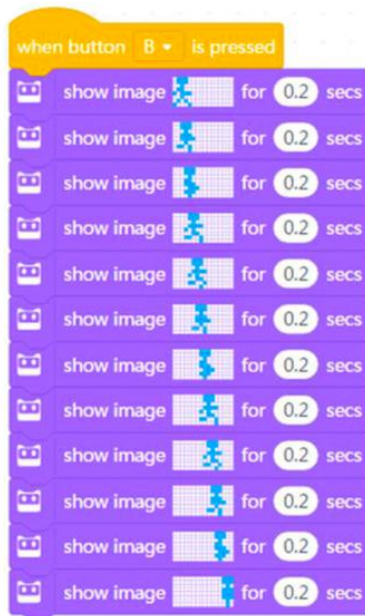


Drugi zadatak: Rastuće drvo





Treći zadatak: Hodanje

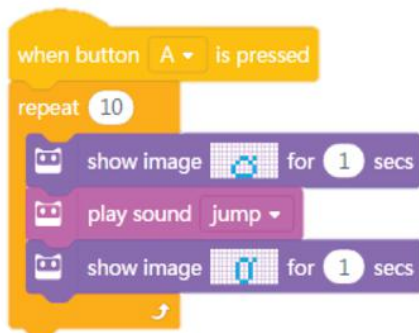


O petljama

U programskim jezicima petlje doživljavamo kao niz uputa koje se trebaju ponavljati. Kada pišemo programe možemo koristiti „Repeat“ blok kako bi zamijenili upute koje se moraju ponavljati, a time će naš program biti uredan i koncizan. Koristeći „Repeat“ blok, možemo napraviti zabavne projekte.

Petlje se koriste za beatove u glazbi i olakšavaju nam igranje igrica. U programiranju se „Repeat“ blok često koristi kako bi kod bio uredniji. Koristeći „Repeat“ blok, Codey Rocky može ponavljati izvađanje sekvence. Kada se „Repeat“ blok jednom iskoristi, brojimo ga kao jedan ciklus ili ponavljanje.

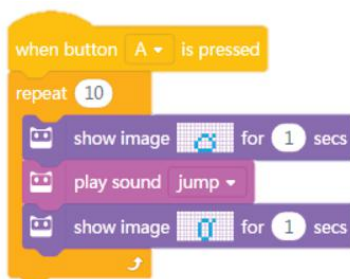
Četvrti zadatak: Kruh ne može skočiti





„Infinite Loop“ je dio koda koji se beskonačno ponavlja. Poblize pogledaj „Forever“ blok. Nema kvrgu na kraju, što znači da ne možeš dodati blok na kraj. To znači da će se „Forever“ blok beskrajno izvoditi. Koristeći „Forever“ blok, možeš napraviti projekte koji su zanimljivi i zabavni.

Peti zadatak: Kruh koji može skočiti



Šesti zadatak: Animacija

Dizajniraj dvije animacije. Svako animaciji daj određeni događaj (pritisni gumb). Za programiranje jedne animaciju koristi „Repeat“ blok, a za drugu koristi „Forever“ blok. Vaše animacije mogu imati priču, jedna može biti „Kruh koji ne skače“ dok druga može biti „Kruh koji skače“, dvije animacije mogu biti međusobno nezavisne.

ZAKLJUČAK

Sekvenca (niz naredbi) se odnosi na niz koraka koje je potrebno poduzeti kako bi se izveo neki zadatak. Petlja u programiranju predstavlja repetitiju određenog dijela koda. „Forever“ blok koristimo za beskonačno ponavljanje koda.

Promisli!



Što sada znam?



Što sam naučio/la?



Što još želim znati?



Codey Rocky vozi sigurno i odgovorno

Što je uvjet?

Donosimo li odluke svakodnevno?

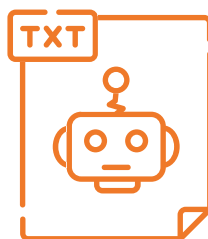
Kakve odluke?

Svaki dan donosimo odluke o onome što ćemo nositi: laganiju ili toplu odjeću. Odluku donosimo na temelju vremenskih uvjeta.

Mogu li roboti donositi odluke?

Mogu li roboti osjetiti svoju okolinu?

U nastavku ćemo upoznati i naučiti što su uvjeti, upoznati i koristiti senzore Codey Rockyja kako bi izvršili određene zadatke.





Uvjet se odnosi na upute koje ovise ako je nešto istinito ili neistinito. Upute će izvršene ako je iskaz istinit, a ako je neistinit, upute neće biti izvršene. U uvjetnom bloku, je heksagonalna rupa, uvjetni kod se stavlja u tu rupu.

Heksagonalni blokovi daju Booleove vrijednost (dvije moguće vrijednosti su istinito i neistinito). Booleova varijabla ima samo dvije vrijednosti, 1 (istinito) i 0 (neistinito). Ako je iskaz istinit Booleova varijabla će imati vrijednost 1 (istinito). U suprotnom slučaju vrijednost će biti (neistino). Dakle, heksagonalne blokove zbog ovih činjenica nazivamo i Booleovim blokovima.

Igra: Uvjetna kutija

Uzmi komadić papira. Na njega napiši uvjetne izjave. Učini svoju izjavu prepoznatljivima. Ovo su neki od primjera: ako imaš dugu kosu; ako nosiš naočale; ako si obučen u crno; ako u imenu imaš slovo „A“ ; ako si rođen u lipnju; ako netko digne ruke.

Presavij papir i stavi ga u kutiju. Iz kutije uzmi list papira. Otvori ga, pročitaj ga naglas i donesi odluku. Ako je uvjetni iskaz istinit, izvrši radnju, a ako nije, preskoči listić. Presavij papir i vrati ga natrag u kutiju. Vrati se na mjesto ili predaj kutiju drugom učeniku.

Prije programiranja moramo locirati senzor za boje

Na prednjem donjem dijelu Rockyja postoji niz senzora: senzor za boje. Grayscale senzor, infracrveni senzor za blizinu i drugi



Infrared Color Sensor

The Color Sensor is in the middle. It can identify multiple colors, like red, green and blue.

Prvi zadatak: Utrka

Kreiraj i testiraj program:

Ako Codey Rocky vidi mahanje zelene zastave, on će startati maksimalnom brzinom.

Kada se pritisne gumb A, Codey Rocky se priprema na startnoj liniji (reproduciraj zvuk).

Ako detektira zelenu boju Codey Rocky će krenuti naprijed punom brzinom.

RGB LED će zasvijetliti crvenom bojom ako program donese odluku koja se temelji na nekoj situaciji.

Ako Codey Rocky identificira crveni predmet, pomaknuti će se nazad.

Infracrveni senzor je u senzoru za boju (lijeva strelica indicira infracrveni primatelj, a desna infracrveni odašiljatelj). S pomoću infracrvenog senzora Codey Rocky može izbjegavati prepreke.

Moraš paziti da senzor za boje bude okrenut na pravu stranu, u suprotnom neće detektirati prepreke.



Drugi zadatak: Izbjegavanje prepreka

Napravi i testiraj program:

Kada će doći do prepreke Codey Rocky će je izbjeći i miče se naprijed.

Stavi neku prepreku ispred Codey Rockyja.

Ako Codey Rocky detektira prepreku i ako se pritisne gumb A, on će se okrenuti za 90 stupnjeva, kreće naprijed, zatim se za 90 stupnjeva okreće lijevo i kreće za naprijed velikom brzinom.

Ako Codey Rocky ne može detektirati nešto ispred sebe, kretati će se naprijed najvećom brzinom.

Dopusti Codey Rockyju da ima izraze lica i da proizvodi zvuk kada dođe do prepreke.

Na donjem desnom kutu Codeyja je crna točkica. Crna točka je senzor za svjetlo. Koristi se za mjerenje intenziteta svjetlosti u njegovom okruženju. Na sredini Codeya možeš naći RGB indikator. Može sijati u bojama.



Na dva načina možemo pristupiti vrijednosti senzora za svjetlo:

Kada je uređaj spojen, stavite kvačicu na blok „ambient light intensity“ kako bi prikazali vrijednost.



Osim toga možete vidjeti skriptu prikazanu ispod i njome prikazati vrijednost na Codeyjevom LED ekranu. Programe koji su prikazani iznad učitaj na Codeyja. Primjeti kako se vrijednost mijenja u obje spomenuti instance.



```

when Codey starts up
  forever
    show ambient light intensity until done
    wait 0.5 seconds
  
```

Mi svakodnevno uspoređujemo vrijednosti. Odluke temeljimo na rezultatima usporedbe. Različite odluke vode prema različitim radnjama. Na primjer, "Ako je moja temperatura 38 stupnjeva, moram ići liječniku." Temeljeći se na vrijednosti termometra mi donosimo odluku ako je potrebno ići liječniku.

Slično tome možemo koristiti „Comparison Operators“ u mBlock 5 kako bi usporedili dvije vrijednosti. Operatori za usporedbu uključuju: < operator, = operator i > operator.

Comparison Operators	Implication	Example
	Less than	The ambient light intensity value is less than 20.
	Equality	The ambient light intensity value equals 20.
	Greater than	The ambient light intensity value is greater than 20.





Treći zadatak: Tunel

Kreiraj i testiraj program:

Kada ulazi u mračan tunel, Codey Rocky pali svjetla i usporava.

Stavi komadić crnog papira na senzor za svjetlo Codey Rockyja.

Kada se pritisne gumb A, Codey Rocky će se kretati naprijed najvećom brzinom.

Ako je intenzitet svjetla manji od 20, Codey Rocky će uključiti svoj bijeli RGB indikator i sporije će se voziti.



ZAKLJUČAK

U programskim jezicima, uvjeti su naredbe čije izvršavanje ovisi o istinitosti ili neistinitosti tvrdnje.

Promisli!



Što sada znam?



Što sam naučio/la?



Što još želim znati?



Codey Rocky upoznaje funkcije

Pranje kose ima tri koraka: šamponiranje kose, masaža vlasišta kako bi napravili pjenu i na kraju ispiranje pjene. Ali što ako ne iskoristimo pojam „pranje kose“ kako bi opisali ovaj niz koraka, kakva će onda biti situacija?

Mi svakodnevno nazivamo radnje i taj se naziv referira na cijeli niz koraka koji su potrebni da se radnja izvrši.

Kada cijelom nizu radnji damo naziv poput "prati kosu", situacija je slijedeća:

Kada te prijatelji pozovu van, a ti još moraš oprati kosu, reći ćeš: "Idem oprati kosu. Čekajte."

Koristimo jednostavnu frazu za cijeli niz radnji. To nazivamo funkcijom.

U programiranju, funkcija se odnosi na niz naredbi u kodu. Kako bi napravili funkciju, prvo joj moramo dati ime. Idući korak je definiranje funkcije, a to se može napraviti tako da u nju dodajemo naredbe.

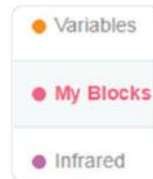
U nastavku ćemo naučiti što su funkcije i definirati ih dodavanjem naredbi.





Napravi funkciju za pokretanje Codey Rockyja i pripazi da se funkcija automatski pokrene kada se Codey Rocky upali.

Otvori mBlock 5 i spoji Codeyja sa softwareom. Prati upute i izvrši izazov. Klikni na „My Blocks“ u traci s kategorijama i odaberi „Make a Block“.



Zatim „definiraj“ blok za pokretanje koji će se pojaviti u području za skripte.



Keje naredbe se trebaju izvršiti kada se Codey pokrene? Dizajniraj programe unutar bloka za pokretanje.



Nakon definiranja funkcije možete je direktno koristiti tako da dodate blok za pokretanje ispod bloka za događaj kada se Codey pokreće.



Unaprijedi ogledni projekt. Možeš promijeniti animaciju ili zvuk.

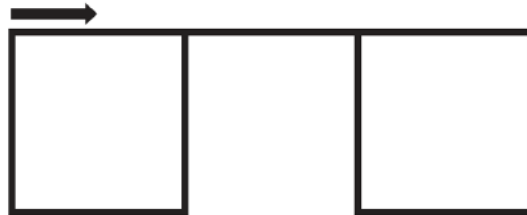
Odaberi jednu od ponuđenih slika i preradi je po želji. Koristeći sliku napravi svoju animaciju.



Drugi zadatak:

Zamisli da je Codey Rocky zaštitar. Patrolira hodnicima zgrade kako bi zaštitio svu imovinu. Trenutno patrolira na prvom katu.

Kreiraj program u kojem Codey Rocky prati crne linije kao što je prikazano ispod.



Napravi funkciju i daj joj naziv „kvadrat“.



Vjerojatno ćeš trebati koristiti sljedeće blokove.



Probaj procijeniti koliko Codeyju treba vremena da prođe linije u obliku kvadratića i linije koja ih povezuje.

- Izmjeri dužinu jedne stranice kvadrata.
- Izmjeri dužinu pravca koji ih povezuje.
- Izračunaj koliko vremena je potrebno da se Codey prođe kvadrat .
- Izračunaj koliko vremena je potrebno da se Codey prođe liniju koja spaja kvadrate.
- Izračunaj Codeyevu brzinu (u sekundama). (Na primjer, ako je Codey Rocky programiran da se kreće naprijed brzinom na 50% snage za jednu sekundu, koja je onda udaljenost vožnje?)

Iskoristi funkciju kvadrat dva ispod bloka za događaj kada se pritisne „A“.



- Codey koristi DC (istosmjerni) motor. Budući da Codey ne može najpreciznije pratiti linije i skretati, u redu je ako otprilike prati put.
- Codey Rocky je programiran da se kreće prema naprijed i da skreće desno. Nakon što ponovi to ponašanje četiri puta, vratiti će se na početnu poziciju.
- Budući da se na putu nalaze dva kvadrata, potrebno je koristiti funkciju kvadrat dva puta.
- Ima više načina prema kojima možemo osigurati Codeyju da prati put kao što je prikazano. Možete probati samostalno shvatiti što je potrebno napraviti ili dovršite zadatak koristeći pseudokod:

Bottom square	Upper square	When button A pressed
move forward 1 second turn right by 90°	move forward 1 second turn left by 90°	bottom square move forward 1 second upper square move forward 1 second
Repeat four times	Repeat four times	Repeat two times



ZAKLJUČAK

U programiranju, funkciju definiramo kao posebno programiran blok. Funkcija se odnosi na niz naredbi koje se mogu koristiti i ponavljati.

Promisli!



Što sada znam?

Što sam naučio/la?

Što još želim znati?

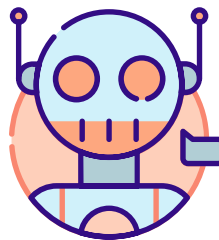


Kamen, šakre i papir!

Zamisli da je varijabla kutija. U kutiju možemo staviti neku vrijednost i zamijeniti je kada god to poželimo. Kutija funkcionira kao alat koji prati rezultat za vrijeme natjecanja. Tijek trajanja natjecanja, rezultati se mijenjaju. Važno je pratiti rezultat i mijenjati vrijednosti.

U računalnom programiranju, varijabla je lokacija za pohranjivanje koja je spojena s odabranim imenom. Kada se varijabla kreira, računalo će ju pohraniti na posebnu lokaciju i toj će lokaciji dati simbolično ime. Koristeći ime te varijable, programi mogu učitati i mijenjati vrijednost u lokaciji gdje je varijabla pohranjena.

U nastavku ćemo upoznati i koristiti varijable.





Igrajmo igru: Vjeveričja kutija

Postoji malena vjeverica. Ima lješnjake u kutiji. Vjeverica svakodnevno u kutiju stavlja lješnjake, ali ih i uzima. Jedan se dan vjeverica probudila i kutiji našla samo 10 lješnjaka. Broj lješnjaka ovisi o različitim situacijama.

Na primjer, vjeverica je bila jako gladna i pojela je 2 lješnjaka, ostalo ih je samo 8. Vjeverica je zatim pronašla 5 lješnjaka i sada ih je 13. Sada ćemo igrati igru i moramo doći do točnog broja lješnjaka.

In the morning, the little squirrel eats 2 nuts for breakfast.

If the maximum temperature exceeds 5°C, the squirrel will pick 10 nuts in the wild.

At noon, the squirrel eats 3 nuts for lunch.

In the afternoon, a little bird pays a visit and gives the squirrel 5 nuts.

Today is the monkey's birthday. The squirrel gives him 3 nuts as a gift.

In the evening, a rat steals 4 nuts.

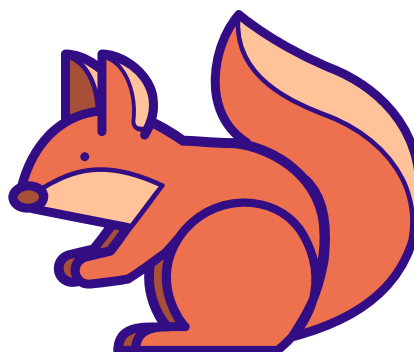
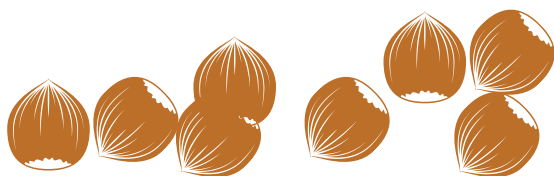
The squirrel plays the rock-paper-scissors game with a friend three times. If he loses the game, he gives away a nut; if he wins the game, he gets a nut; if it's a draw, then no gain or loss for both sides.

In the evening, the squirrel eats only 1 nut to keep fit.

It's hot today. The squirrel buys a straw hat from Aunt Bear. It costs him 7 nuts.

If the amount of the nuts in the box is less than 5, the squirrel will pick 10 nuts outdoors.

If the amount of the nuts in the box exceeds 12, the little squirrel will turn in a circle happily.





Koja je varijabla u ovoj igri? Kako ste ju nazvali?

Koja je početna vrijednost varijable?

Koje operacije smo izveli s pomoću varijable?

Prvi zadatak:

Koristeći varijablu za dodjeljivanje vrijednosti možemo odrediti brzinu kretanja Codey Rockyja. Kada se pritisne gumb A, brzinu stavljamo na 30 i Codey Rocky će se kretati naprijed postavljenom brzinom jednu sekundu.



Codey Rocky stoji na pozornici i pozdravlja publiku.

Codey Rocky se okreće 70 stupnjeva lijevo, a zatim 140 stupnjeva desno. Na kraju se vraća na početnu poziciju. Kako bi se Codey okretao pod određenim kutovima, potrebno je napraviti varijablu "kut".

Postavi varijablu kut na vrijednost od 70.



Neka se Codey okrene lijevo pod zadanim kutom.



Varijabli „kut“ dodaj novu vrijednost. Kao što je programirano, Codey okrenuti desno pod kutom od 140 stupnjeva, zatim lijevo pod kutem od 70 stupnjeva.

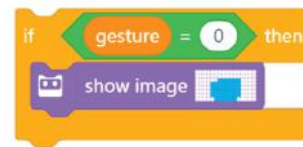
to Codey Rockyiju dodaj izraze lica, zvukove i svjetla.

Drugi zadatak: Kamen, škare i papir

Možete igrati kamen, škare i papir koristeći Codey Rockyja.

Kada se Codey pokrene, sve početne vrijednosti će biti postavljene na 0.

Codey Rocky mijenja gestikulacije nasumično. Brojevi 0, 1 i 2 označuju kamen, škare i papir. Kada se Codey potrese, varijabla "gesta" će nasumično odabrati jedan od tri broja. Ako je vrijednost 0, LED ekran će prikazati prizor šake. Ako je vrijednost 1, LED će prikazati škare. Ako je vrijednost 2, LED će prikazati papir.



Ako Codey pobijedi, potrebno je pritisnuti gumb A i varijabla "pobjeda" će promijeniti vrijednost za 1. U međuvremenu Codey prikazuje nasmejano lice i reproducira zvuk „smijeh“.

Ako Codey izgubi potrebno je pritisnuti gumb B i varijabla "poraz" će promijeniti vrijednost za 1. U međuvremenu Codey prikazuje tužno lice i reproducira zvuk „tužan“.

Ako je neriješeno, pritisni gumb C i varijabla "neriješeno" će promijeniti vrijednost za 1. U međuvremenu, Codey izgleda mirno i proizvodi žamor.

U nekim će slučajevima, Codey će pogledati koja mu je šansa za pobjedu. Kada je intenzitet svjetla manji od 2, šansa za pobjedu će biti prikazana na LED ekranu u obliku decimala.



**Bilješke:**

Kada se Codey programira potrebno je koristiti do 6 događaj blokova.

Kako bi poboljšali vjerojatnost pobjede, kodu je potrebno dodati operator blokove.

Formula: Pobjede/pobjede+ porazi + neriješeno = Šansa pobjede



Kada se pritisne gumb A, Codey emitira infracrveni signal; ako drugi Codey primi signal, njegovi poeni se smanjuju za 1; igra završava kada poeni dođu do 0.

Codeyjeve uši imaju infracrvene odašiljače i primatelje, koji omogućuju bežičnu komunikaciju između dva Codeya

IR Transmitter

Facilitate the communications between Codeys

**IR Receiver**

Receive the signals from the IR transmitter



Kada se Codey pokrene, imat će 10 poena. Idući korak je korištenje „forever“ bloka kako bi detektirali ako je protivnik „upucao“ Codeya. Ako Codeya upucaju, vrijednost poena se mijenja za "-1" i reproducira zvuk "iznenađenja". Igra će biti gotova kada poeni dosegnu nulu i kada će Codey izgledati tužno. Ako je Codeyju ostalo nekoliko poena (vrijednost je veća od 0) i kada se pritisne gumb A poslati će infracrvenu poruku i reproducirati zvuk „laser“.

ZAKLJUČAK

U programiranju, varijabla je spremnik u koji možemo pohraniti informacije. Varijabla se može duplicirati, informacije u njoj se mogu promijeniti i varijable možemo uspoređivati. S pomoću „operators“ blokova, možemo koristiti varijable kako bi pohranili rezultate računskih operacija.



<http://erasmus-artie.eu>

Odaberi blok „send IR message“ u kategoriji "Infrared". Neka si dva Codeya šalju različite poruke. Na ovaj se način mogu identificirati. Na primjer, jedan Codey šalje poruku "crveno" a drugi Codey šalje poruku "plavo". U ovom slučaju, poruke su metci.



Kada Codey A primi poruku od Codeya B, Codey A je upucan i poeni mu se smanjuju za 1 (možete vidjeti ako je upucan na „statement“ bloku). Nakon što se Codeya upuca, on će reproducirati zvuk iznenađenja i izgubiti jedan poen.

Promisli!

Što sada znam?

Što sam naučio/la?

Što još želim znati?