|  |
| --- |
| **Računalno razmišljanje za više razrede osnovne škole: upoznavanje s algoritmima** |

**Procijenjeno trajanje:** 1 sat

**Dobna skupina:** učenici viših razreda osnovne škole

**Ciljevi učenja, vještine i kompetencije**

Glavni je cilj upoznati učenike s računalnim razmišljanjem te im pomoći u razvoju vještina matematičkog rasuđivanja. Cilj je sustavno razmatrati neki problem i razmisliti kako bi se rješenje nekog problema moglo prenijeti u računalni kod. Učenici će se upoznati s pojmom algoritama.

**Aktivnosti i uloge**

Učenici će igrati igre s brojevima i odgovarati na pitanja koja potiču njihovo matematičko rasuđivanje. Za ovu će aktivnost raditi u parovima. Glavna je uloga nastavnika da daje upute, prati razred tijekom izvođenja aktivnosti i pruža pomoć kad je potrebno. Povratne informacije mogu se dati tijekom aktivnosti i na kraju.

**Što vam je potrebno?**

Za ovu aktivnost vama i vašim učenicima trebat će samo interaktivna bijela ploča ili projektor s pristupom internetu kako biste predstavljali i pretraživali aktivnosti u alatu Scratch, prijenosna ili stolna računala za igranje igre u alatu Scratch te olovke i papir.

**Prostor za učenje**

Školska ili računalna učionica.

**Opis aktivnosti**

Odigrajte inačicu igre „Pogodi broj” s učenicima u razredu. Recite im da ste zamislili broj od 0 do 127 i pozovite ih da ga naizmjence pogađaju. Svaki im put recite je li njihov broj previsok, prenizak ili točan. Nakon svakog koraka na ploču zapišite predloženi broj te je li previsok ili prenizak.

Neka učenici razmisle o tome koliko su njihovi pokušaji pomogli razredu u tome da brzo pronađe nepoznati broj. To može poslužiti kao primjer. Recimo da razred zna da je broj 64 ili veći; ako sljedeći igrač predloži broj 96, prepolovit će raspon mogućnosti. Ako predloži broj 65, vjerojatno neće dobiti mnogo više informacija o točnom broju.

Neka učenici u parovima igraju igru jedan protiv drugog i pokušaju pronaći dobru strategiju kako bi pobijedili u što manje koraka. Nakon što su učenici dobili priliku za vježbu, zamolite ih da svoju strategiju objasne razredu kako bi ostali učenici mogli komentirati učinkovitost predloženih pristupa.

Pokažite učenicima program u alatu Scratch, na <https://scratch.mit.edu/projects/240290316/>, i dajte im vremena da zajedno s partnerom dokuče što program radi *bez* njegova pokretanja. Može li neki par objasniti program razredu? Program samo preuzima ulogu nastavnika iz prve inačice igre: bira broj i zatim na svaki pokušaj odgovara s „prenizak”, „točan” ili „previsok”. Objasnite im da ste se prethodno služili algoritmom i da taj program jednostavno upotrebljava isti algoritam koji ste vi slijedili, ali u obliku koda. Moglo bi biti korisno učenicima pokazati algoritam u obliku dijagrama toka.

Neka učenici pokušaju razraditi najučinkovitiju strategiju za pobjedu u što manje koraka. Na što moraju obratiti pažnju u svakom koraku? Kako bi trebali odabrati broj koji će predložiti? Ako predlože prenizak broj, što bi iduće trebali predložiti? Ako je broj previsok, kako da nastave pogađati?

Zamolite učenike da se dosjete kako bi programirali Scratch da brzo pogodi *njihov* broj, a ne obratno. Ako učenici dosad nisu mnogo programirali u alatu Scratch, možda će htjeti istražiti nedovršenu inačicu na <https://scratch.mit.edu/projects/240290555/>, no ako su već prilično vješti u programiranju u tom alatu, zajedno s partnerom mogu isprogramirati nešto slično. Na <https://scratch.mit.edu/projects/240290793/> može se pronaći dovršena inačica.

Ako učenici dovrše zadatak, izazovite ih da prilagode kod na način da prati broj unesenih pokušaja. Mogu li učenici izmijeniti kod kako bi vidjeli koliko brzo Scratch može pogađati brojeve manje od tisuću? Ili brojeve manje od milijun? Dodatno bi mogli prilagoditi kod kako bi se ažurirao popis pogođenih brojeva i odgovora igrača u svakom koraku.

Na kraju istražite kako su učenici u okviru aktivnosti iskoristili koncepte računalnog razmišljanja: morali su logički razmisliti o tome koji su brojevi preostali u svakom koraku i kako bi Scratch trebao odgovoriti na unose igrača. Program se temelji na klasičnom algoritmu za pretraživanje – binarnom pretraživanju. Izvorni problem pogađanja brojeva od 0 do 127 rastavljamo na niz postupno manjih problema (poznato kao algoritam „podijeli pa vladaj”). Problem se sastoji od najmanjeg i najvećeg preostalog broja i ta je apstrakcija zapravo jedino što program mora pratiti u svakom koraku.

**Ime autora: Miles Berry**