

Gra planszowa MOON czyli informatyka bez komputera

Anna Grzybowska, Elżbieta Kawecka, Witold Kranas

50 lat temu człowiek po raz pierwszy stanął na Księżycu. 20 lipca 1969 roku o godzinie 20:17:40 UTC po czterech dniach lotu misji Apollo 11, Neil Armstrong i Buzz Aldrin w module księżycowym Eagle bezpiecznie wylądowali na terenie Morza Spokoju na Księżycu. Lądowanie nie obyło się jednak bez problemów. Eagle przeprowadzał lądowanie na podstawie ciągłych obliczeń komputera pokładowego. Jednakże niecałe trzy minuty przed lądowaniem komputer zarejestrował kilka alarmów: radar, który powinien być wyłączony podczas procedury lądowania, niespodziewanie włączył się. Astronauci zostali zaalarmowani komunikatem: błąd 1202, oznaczającym brak wolnej pamięci. Komputer pokładowy, który przeprowadzał manewr lądowania, mógł zostać przeciążony przez dodatkową pracę. Na szczęście oprogramowanie zarządzające komputerem, zaprojektowane przez zespół inżynierów z MIT kierowany przez Margaret Hamilton było wystarczająco dobre, aby dać sobie radę z tym problemem i skupić całą moc obliczeniową na manewrze lądowania. Steve Bales, specjalista komputerowy z centrum lotów zdecydował, że astronauci mogą kontynuować lądowanie.



Rysunek 1. LGC – komputer lądownika Eagle

Komputer lądownika Lunar Guidance Computer (LGC) został zaprojektowany w 1966 w MIT. Miał on moc obliczeniową taką jak współczesne programatory pralek automatycznych albo kalkulatory. Posługiwał się słowem 16-bitowym, był sterowany zegarem o podstawowej częstotliwości taktowania 2048 kHz, miał masę 32 kilogramów, korzystał z ferrytowej pamięci stałej o pojemności 74 kB i nietrwałej o pojemności 4 kB. Jak wynika z informacji Google'a, jedno żądanie wyszukiwania w przeglądarce potrzebuje więcej mocy obliczeniowej, niż cały 11-letni program Apollo, w czasie którego przeprowadzono kilka misji bezzałogowych, 11 załogowych i sześć lądowań na Księżycu.

Projekt COMPUS

Historia lądowania na Księżycu była inspiracją do stworzenia gry planszowej dla uczniów w ramach projektu COMPUS - Komputer to my. Pomysłodawcami i koordynatorami projektu są pracownicy naukowcy Uniwersytetu Deusto w Bilbao, a partnerami szkoły z Hiszpanii i Rumunii, firma AGR Priority z Hiszpanii oraz Ośrodek Edukacji Informatycznej i Zastosowań Komputerów w Warszawie. Projekt COMPUS, realizowany w ramach grantu Erasmus+, jest zaplanowany na dwa lata (1.09.2018-31.08.2020). W ramach projektu zostaną opracowane i udostępnione dwie gry planszowe wraz z obudową metodyczną.



Rysunek 2. Partnerzy projektu COMPUS przed siedzibą OEliZK w Warszawie

Pierwsza gra (MOON) jest już gotowa i można ją bezpłatnie pobrać ze strony¹. Znajdują się tam karty gry do wydruku oraz instrukcja w wielu językach. Dostępne są wersje w języku angielskim, hiszpańskim, polskim, rumuńskim, baskijskim i katalońskim, opracowane przez partnerów projektu. Dodatkowo instrukcja gry została przetłumaczona przez wolontariuszy na język portugalski, francuski, niemiecki i duński.

Obecnie trwają prace nad koncepcją drugiej gry, nawiązującej również do idei myślenia komputacyjnego oraz nad poradnikiem dla nauczycieli. Całość materiałów będzie dostępna wiosną 2020 roku.

Wspólna praca uczestników projektu jest okazją do wymiany pomysłów i wzajemnej inspiracji. Dotychczas miały miejsce trzy spotkania projektu. Pierwsze z nich odbyło się w formie wideokonferencji i miało na celu wzajemne poznanie się oraz omówienie zasad pracy w projekcie. Kolejne spotkania odbyły się w Warszawie i Bukareszcie. Uczestnicy testowali różne gry planszowe pod kątem ich przydatności w procesie uczenia się oraz atrakcyjności i potencjalnego zainteresowania wśród uczniów.



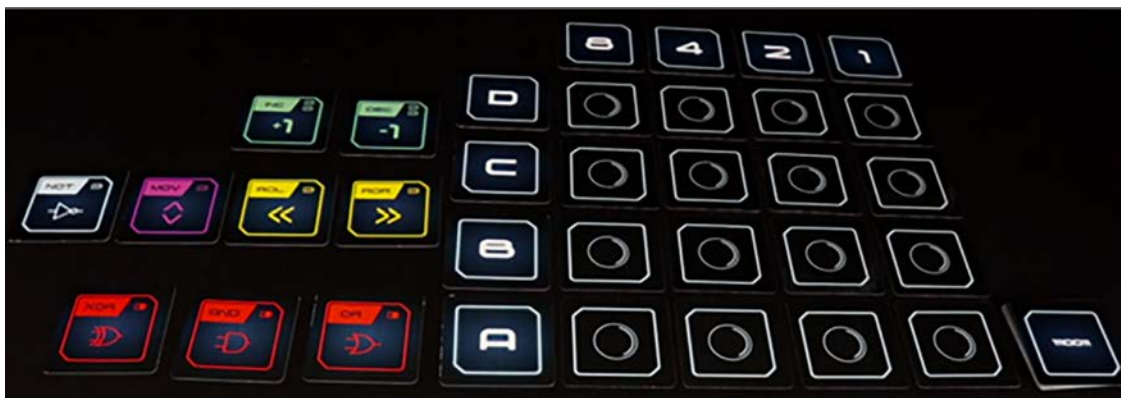
Rysunek 3. Testowanie gry MOON podczas spotkania w Warszawie

¹ <http://compus.deusto.es/moon>

Gra MOON

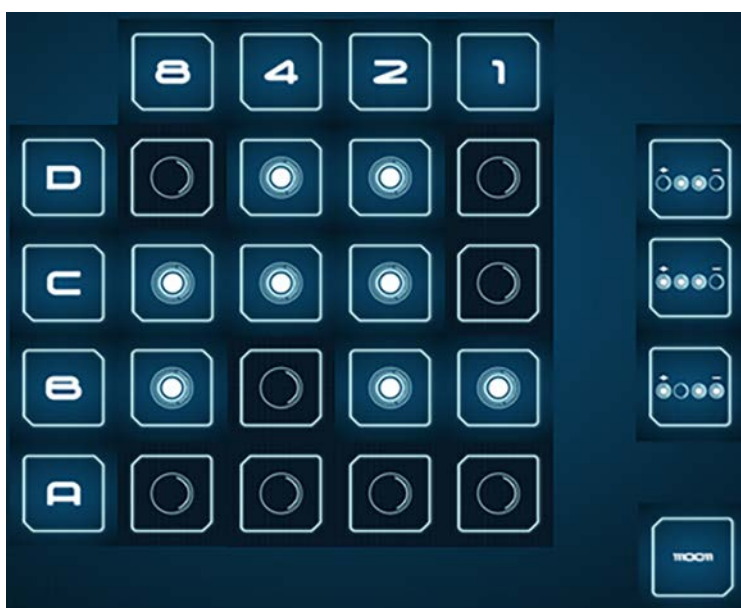
MOON to edukacyjna gra planszowa, w której uczestnicy sterują pracą prostego komputera, przeprowadzając operacje na jego procesorze. Przy okazji uczą się liczyć w systemie dwójkowym, wykonywać operacje logiczne i dowiadują się, jak działa komputer. Gra jest przeznaczona dla nastolatków (od 10 lat), może w niej brać udział od 1 do 4 graczy. Czas trwania rozgrywki wynosi około 15 - 45 minut w zależności od wybranego poziomu trudności.

Przygotowanie gry – układanie planszy



Rysunek 4. Podstawowa plansza gry MOON

- Umieszczamy cztery rejestry (A, B, C i D) procesora czyli CPU i odpowiadające im karty z wyłączonymi bitami pośrodku stołu. To będzie plansza centralna.
- Po lewej stronie planszy centralnej umieszczamy karty operacji. Niech będą one uporządkowane według zużycia energii: najpierw te, które wymagają 2 jednostek energii (INC, DEC), a następnie te, które wymagają 1 jednostki energii (NIE, ROL, ROR, ROL, MOV), i wreszcie te, które wymagają 1/2 jednostki energii (OR, AND, XOR).
- Tasujemy karty zadań i kładziemy zakrytą talię po prawej stronie planszy. Te karty reprezentują obliczenia, które astronauta muszą wykonać, aby wylądować na Księżycu.
- Bierzemy pierwsze trzy karty zadań i układamy według nich rejestry B, C i D procesora.
- Wszystkie karty potrzebne do gry oraz podręcznik (w polskiej wersji językowej) znajdują się na stronie projektu². Można je wydrukować, odpowiednio przyciąć i zacząć grę.



Rysunek 5. Przykładowy układ procesora na początku gry

² <http://compus.deusto.es/moon>

Teraz pora na wykonanie pierwszego zadania, czyli zapełnienie rejestru A procesora według kolejnej karty zadań z talii. Ale zanim je wykonamy trzeba się dowiedzieć czegoś na temat znaczenia bitów procesora i operacji, które można na nich wykonywać.

Bity i operacje

Rejestry procesora oraz karty zadań w naszej grze są czterobitowe. Przykładowo zawartość rejestru przedstawionego na rysunku to $8+4+1=13$.



Rysunek 6. Rejestr procesora zawiera liczbę 13.

Operacje, które można wykonywać na rejestrach procesora przedstawia tabela poniżej. Po prawej stronie obok nazwy operacji widać ile porcji energii trzeba zużyć, aby ją wykonać. Standardowo wykonanie zadania wymaga nie więcej niż trzech porcji energii. Operacje można wykonywać na dowolnych rejestrach procesora.



INC

Dodaje 1 do wartości zapisanej w rejestrze.

Ta operacja jest stosowana w jednym rejestrze i kosztuje 2 jednostki energii. Jeśli rejestr przechowuje wartość maksymalną (wszystkie bity włączone), następuje przepełnienie i rejestr jest resetowany do zera.

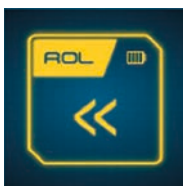


DEC

Odejmuje 1 od wartości zapisanej w rejestrze.

Ta operacja jest stosowana w jednym rejestrze i kosztuje 2 jednostki energii.

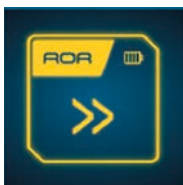
Jeśli wszystkie bity rejestru są wyłączone, odjęcie 1 spowoduje niedobór i ustawienie wszystkich bitów rejestru na jeden.



ROL

Polega na przesunięciu każdego bitu w rejestrze w lewo i umieszczeniu bitu znajdującego się po lewej stronie w skrajnie prawej pozycji.

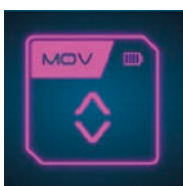
Ta operacja jest stosowana w jednym rejestrze i kosztuje 1 jednostkę energii. W wielu przypadkach jest to równoznaczne z pomnożeniem wartości rejestru przez 2.



ROR

Polega na przesunięciu każdego bitu w rejestrze w prawo i umieszczeniu bitu znajdującego się po prawej stronie w skrajnie lewej pozycji.

Ta operacja jest stosowana w jednym rejestrze i kosztuje 1 jednostkę energii. W wielu przypadkach jest to równoznaczne z dzieleniem wartości rejestru przez 2.



MOV

Kopiuje wszystkie bity z jednego rejestru do drugiego, zastępując wartość przechowywaną w miejscu docelowym.

Ta operacja jest stosowana w dwóch rejestrach lub rejestrze i module pamięci RAM i kosztuje 1 jednostkę energii (1/2 w trybie rywalizacji).



NOT

Negacja każdego bitu w rejestrze: włączone bity zostają wyłączane, a wyłączane bity zostają włączone. Wymaga to odwrócenia wszystkich bitów rejestru.

Ta operacja jest stosowana w jednym rejestrze i kosztuje 1 jednostkę energii.



OR

Kopiuje tylko włączone bity z jednego rejestru do drugiego.

Ta operacja jest stosowana w 2 rejestrach i kosztuje 1/2 jednostki energii.

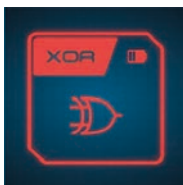
Kopiuje ona tylko włączone bity z jednego rejestru do drugiego.



AND

Kopiuje tylko wyłączane bity z jednego rejestru do drugiego.

Ta operacja jest stosowana w 2 rejestrach i kosztuje 1/2 jednostki energii.



XOR

Kopiuje tylko włączone bity z jednego rejestru do drugiego, ale jeśli bit był już włączony zostaje on wyłączony.

Ta operacja jest stosowana w 2 rejestrach i kosztuje 1/2 jednostki energii.

Przykładowa gra

Przypuśćmy, że przy stanie rejestrów takim jak na rysunku 5 dostajemy kartę zadania z następującym układem bitów:



Rysunek 7. Zawartość przykładowej karty zadania

Możliwe rozwiązanie to:

Użycie operacji **MOV** – przeniesienie bitów z rejestru D do A, na co zużywamy 1 porcję energii.

Zastosowanie w rejestrze A operacji **DEC** – zmniejszenie zawartości rejestru o 1, co kosztuje 2 porcje energii.



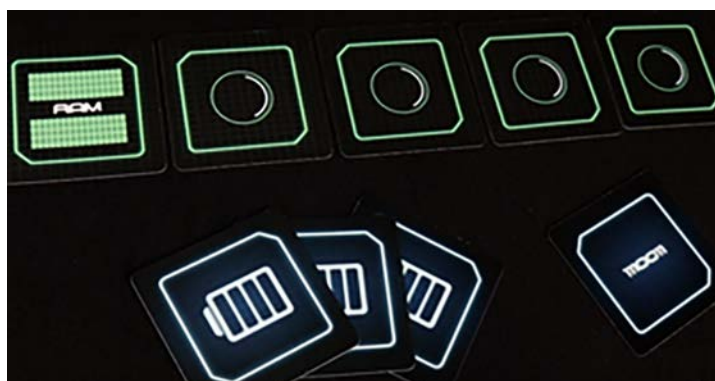
Zadanie udało się wykonać przy użyciu 3 porcji energii. Możemy więc wziąć następną kartę zadania. Jeżeli nie uda się zrealizować zadania za pomocą 3 porcji energii, trzeba zakrytą kartę położyć po prawej stronie planszy i wziąć z talii kolejną kartę zadania. Pięć odłożonych zakrytych kart oznacza przegraną.

Istnieją również karty zadań, które nie mają kombinacji bitów, ale zawierają informację o błędzie (ERROR). Te specjalne karty nie mogą zostać odrzucone i będą blokować jedną z pozycji listy oczekujących zadań przez resztę gry.

Jeśli karta zadania znajdzie się na koniec rundy na piątej pozycji, nasz procesor okazał się zbyt wolny, gra się kończy i nasza misja księżycowa nie powiodła się. Z drugiej strony, jeśli uda się szybko rozwiązać wszystkie zadania ze stosu, astronauta będą mogli bezpiecznie wylądować na Księżycu, a my wygramy!

Gra w trybie rywalizacji

Każdy gracz wybiera kolor, bierze kartę pamięci RAM tego koloru i ustawia wszystkie pozycje swojego modułu RAM na zero. Każdemu graczowi przydziela się jednostki energii zgodnie z wybranym poziomem trudności: 4 dla gry łatwej, 3 dla normalnej, 2,5 dla trudnej i 2 dla mistrzowskiej.



Rysunek 8. Zasoby gracza w grze rywalizacyjnej.

Każdy gracz bierze kartę zadania z talii, patrzy na nią (nie pokazując jej innym graczom) i kładzie ją zakrytą obok swojego modułu pamięci RAM.

Jeśli graczowi podczas swojej tury uda się zapisać swoje zadanie w rejestrze A procesora, pokazuje on swoją kartę zadania pozostałym graczom, umieszcza ją obok swojego modułu pamięci RAM i bierze kolejną kartę zadania z talii.

Po wyczerpaniu talii kart zadań zwycięzcą zostaje gracz, który wykonał najwięcej zadań.

Uwagi końcowe

Więcej informacji o projekcie COMPUS, współfinansowanym przez Program Unii Europejskiej Erasmus+, można znaleźć na stronie: <http://compus.deusto.es/moon>

Wsparcie Komisji Europejskiej dla produkcji tej publikacji nie oznacza poparcia dla treści, które odzwierciedlają tylko poglądy autorów, a Komisja nie ponosi odpowiedzialności za wykorzystanie informacji w niej zawartych.