

JĘZYK STREOWANIA ROBOTEM

PROGRAMOWYM BY ADAM BUCZEK

INSTRUKCJA OBSŁUGI

1. Oznaczenia(plansza):

- ^, <, v, >: oznaczają robota i jednocześnie kierunek w którym się porusza(zgodnie z kierunkiem strzałek)
- +: oznacza że na danym polu znajduje się co najmniej jeden klocek, który może zostać podniesiony przez robota
- x: oznacza ścianą, czyli pole na które robot nie może wejść
- o: oznacza puste pole na planszy

2. Oznaczenie(pliki konfiguracyjne):

- pierwsza linia pliku zawiera rozmiar planszy
- kolejne linie(ilość w zależności od rozmiaru planszy) zawierają stan każdego pola
 - jedna linia to jeden wiersz planszy
 - 0: oznacza puste pole
 - większe od 0 oznacza ilość klocków na polu
 - -1 oznacza ścianę
- kolejna linia zawiera parametry robota oddzielone spacją: kolejno pozycja wertykalna, pozycja horyzontalna, kierunek(0 – góra, 1 – lewo, 2 – dół, 3 – prawo), pojemność kieszeni, zawartość kieszeni
- ewentualne kolejne linie oznaczają makra i numer linie makra w które aktualnie robot wykonywał w czasie zapisywania pliku(format linii NAZWAMAKRA
NRLINII)

3. Obsługa programu

- bez wczytania pliku konfiguracyjnego robot jest ustawiany w lewym górnym rogu planszy, skierowany w dół. Pojemność kieszeni jest ustawiana na 10, a zawartość na 5.
- robot ma wbudowane funkcje
 - move: przesuwa robota o pole do przodu zgodnie z kierunkiem
 - turnLeft: obraca robota w lewo
 - take: bierze z pola klocek jeżeli taki jest i robot ma miejsce w kieszeni
 - put: odkłada klocek na pole jeżeli ma jakiś w kieszeni
 - isWall: sprawdza czy przed robotem jest ściana
 - isBrick: sprawdza czy na polu na którym znajduje się robot jest klocek
- możliwe jest również definiowanie własnych funkcji(umieszczane w katalogu ROBOT/functions z rozszerzeniem txt) na podstawie języka sterowania

4. Język sterowania

- słowa kluczowe: procedure, while, if, begin, end, not

- przykładowa poprawna funkcja o nazwie going.txt

```
procedure going
```

```
begin
```

```
    while not isWall
```

```
    begin
```

```
        move
```

```
        if isWall
```

```
        begin
```

```
            turnRight
```

```
        end
```

```
    end
```

```
end
```

aby była poprawna musi istnieć funkcja turnRight(zdefiniowana przez użytkownika)

5. Wczytywanie stanu

6. Zapis wyglądu planszy do pliku

- jeżeli nie włączy zapisu krokowego do pliku, program automatycznie zapisze wygląd planszy po każdej interpretacji funkcji podanej przez użytkownika do pliku o nazwie DZIEŃ-MIESIĄC-ROK-GODZINA-MINUTA-SEKUNDA(ustawione na początku programu), w przeciwnym razie każdy krok będzie zapisywany do pliku który poda użytkownik. Katalog w którym znajdują się te pliki ROBOT/savedFiles

- w programie istnieje możliwość wczytania stanu planszy, robota oraz ewentualnego stanu wykonywanych funkcji, wczytanie następuje z pliku konfiguracyjnego z katalogu ROBOT/configurationFiles

7. Wykonanie krokowe

- w programie istnieje możliwość pracy krokowej, tzn. wykonanie jednego kroku z makra i oczekiwanie na reakcję użytkownika, użytkownik w tym momencie może przerwać wykonania, wykonać całe makro, wykonać kolejny krok, wykonać inne makro, zapisać stan lub wczytać stan.