

Symulacja gry w Piotrusia (w wersji dla 2 graczy: **g1** oraz **g2**).

Idea gry

Gra składa się z talii kart, w skład talii wchodzi **pary** z nazwami **zwierzaków** (zwykle z obrazkami) oraz specjalna karta **piotrek**, która jest bez pary.

Celem gry jest kompletowanie i wyrzucanie par przez graczy. Gracz, który pozostanie z kartą **piotrek** przegrywa (wygrywa gracz, który jako pierwszy pozostanie bez kart).

1. Generujemy **talie** kart na podstawie podanego opisu kart (patrz później)
2. Grę rozpoczynamy **tasując** karty i **rozdając** je na przemian **2** graczom
3. Na początku gry gracze **usuwiają pary** z kart posiadanych przez siebie po rozdaniu
4. Gracz **g1** rozpoczyna rozgrywkę (a ogólnie: gracz z **mniejszą** liczbą kart), czyli wyciąga kartę od przeciwnika **g2**
Jeśli **gracz** wyciągnie kartę pasującą do pary z którąś z posiadanych kart, wówczas usuwa tę parę.
Jeśli wyciągnie kartę **piotrek**, bierze go do siebie (tj wstawia na koniec).
W kolejnym ruchu przeciwnik powtarza to samo, itd.
5. Gra kończy się w momencie, gdy któryś z graczy pozostanie z kartą **piotrek**

Realizacja zadania

Funkcja **main** została przygotowana w całości i nie należy jej modyfikować, poza komentowaniem lub odkomentowywaniem kolejnych etapów zadania. Niektóre funkcje są przygotowane.

W całym zadaniu implementowanym samodzielnie, nie wolno zliczać elementów listy (do przeglądania listy działamy na wskaźnikach lub korzystamy z pola ogon).

W całym zadaniu należy pilnować poprawności pól: głowa oraz ogon.

Nie tworzymy list „pomocniczych”, działania należy wykonać bezpośrednio w podanej liście.

Etap 1 (2 pkt)

Należy zaimplementować:

gotowa (tylko obejrzyj)

- `int init(gracz* g, char* karty_startowe[N])`
generuje talie kart (w strukturze wskazywanej przez g wypełnia jej pola) zawierającą talie kart (**pary + piotrek**) na podstawie przekazanej tablicy nazw **karty_startowe**
zwierzaki wstawiamy 2-krotnie, a kartę **piotrek** - pojedynczo.

Funkcja **init** wymaga zaimplementowania funkcji **tworz_karte** oraz **wstaw**

0,5 pkt

- `karta* tworz_karte(char* nazwa);`
tworzy dynamicznie kartę o podanej nazwie (alokujemy pamięć na kartę oraz jej nazwę), zwraca NULL gdy nie udało się utworzyć.

0,5 pkt

- `int wstaw(gracz* g, karta* k);`
wstawia kartę k **na koniec listy** kart gracza; zwraca 1, gdy ok., wpp zwraca 0

ponadto zaimplementuj:

0,5 pkt

- `void wypisz(gracz g);`
która **wypisuje** karty na ekran gracza g

0,5 pkt

- `void zwolnij(gracz* g);`
która **zwalnia pamięć** zaalokowaną dla gracza g

Przykład utworzonej i wypisanej talii kart (cd wyników na końcu).

KARTY:

zyrafa zyrafa papuga papuga lew lew kot kot pies pies tygrys
tygrys pantera pantera motyl motyl goryl goryl puchacz puchacz
krokodyl krokodyl kaczka kaczka szczur szczur mysz mysz bocian
bocian krecik krecik piotrek

Etap 2 (1,5 pkt)

Należy zaimplementować:

1 pkt

- **void** tasuj(gracz* g);

Algorytm: dla każdej kolejnej karty losuj `j = rand()%8`, jeśli `j=1,2,3` to **przenosimy** kartę **na początek listy** kart (ale „głowa” jest już na początku, czyli zostaje), dla pozostałych wartości (tj. 0, 4, 5, 6, 7) przechodzimy do kolejnej karty (bez zmian). Do testowania wartości `j` należy zastosować instrukcję **switch**

0,5 pkt

- **void** rozdaj(gracz g, gracz* g1, gracz* g2);

„rozdaje” karty 2 graczom: reprezentowanym wskaźnikami **g1** oraz **g2** (**przenosząc** naprzemiennie kolejne karty z głowy **g** po jednej, pierwszą kartę zawsze dostaje **g1** – zawsze na koniec listy). Do wstawiania wywołaj funkcję **wstaw**.

Przypomnienie: nie wolno zliczać liczby kart.

Etap3 (1,5 pkt)

- **void** usuwaj_pary(gracz* g);

Należy **usunąć** graczowi karty, które tworzą **parę** (pozostałe karty bez zmian).

Przypomnienie: działamy bezpośrednio wskaźnikami w liście, nie tworzymy list pomocniczych.

Etap4 (2 pkt)

1 pkt

- **void** usuwaj_karte(gracz* g, char* nazwa);
usuwa graczowi jedną kartę (pierwszą) o podanej nazwie (jeśli występuje)

1 pkt

- **void** wyciagnij_karte(gracz* g1, gracz* g2);

Algorytm:

Gracz **g1** „wyciąga” od **g2** jego pierwszą kartę (tj. z głowy)

- wypisz wylosowaną kartę
- jeśli wyciągniętą kartą jest **piotrek**, należy usunąć kartę z **g2** i wstawić ją do **g1** - na koniec. Wywołaj **wstaw** (oraz pole **ogon**)
- jeśli wyciągniętą kartą - nie jest **piotrek**, to należy usunąć ją obu graczom. Wywołaj **usuwaj_karte**.

Uproszczona **rozgrywka gry w Piotrusia** odbywa się za pomocą funkcji:

```
int gra(gracz* g1, gracz* g2);
```

która jest już napisana ☺... i zwraca kto wygrał....

Powodzenia !

KARTY:

zyrafa zyrafa papuga papuga lew lew kot kot pies pies tygrys tygrys pantera
pantera motyl motyl goryl goryl puchacz puchacz krokodyl krokodyl kaczka kaczka
szczur szczur mysz mysz bocian bocian krecik krecik piotrek

POTASOWANE:

krecik mysz krokodyl puchacz pies pies pantera motyl puchacz piotrek szczur
szczur krokodyl goryl motyl tygrys kot lew lew papuga zyrafa zyrafa papuga kot
tygrys pantera goryl kaczka kaczka mysz bocian bocian krecik

KARTY G1

krecik krokodyl pies pantera puchacz szczur krokodyl motyl kot lew zyrafa papuga
tygrys goryl kaczka bocian krecik

KARTY G2

mysz puchacz pies motyl piotrek szczur goryl tygrys lew papuga zyrafa kot
pantera kaczka mysz bocian

KARTY G1 (po USUWANIU par)

pies pantera puchacz szczur motyl kot lew zyrafa papuga tygrys goryl kaczka
bocian

KARTY G2 (po USUWANIU par)

puchacz pies motyl piotrek szczur goryl tygrys lew papuga zyrafa kot pantera
kaczka bocian

STARCIE 1

- wyciągnięto: puchacz

karty G1: pies pantera szczur motyl kot lew zyrafa papuga tygrys goryl kaczka
bocian

karty G2: pies motyl piotrek szczur goryl tygrys lew papuga zyrafa kot pantera
kaczka bocian

STARCIE 2

- wyciągnięto: pies

karty G1: pantera szczur motyl kot lew zyrafa papuga tygrys goryl kaczka bocian

karty G2: motyl piotrek szczur goryl tygrys lew papuga zyrafa kot pantera kaczka
bocian

STARCIE 3

- wyciągnięto: motyl

karty G1: pantera szczur kot lew zyrafa papuga tygrys goryl kaczka bocian

karty G2: piotrek szczur goryl tygrys lew papuga zyrafa kot pantera kaczka bocian

STARCIE 4

- wyciągnięto: pantera

karty G1: szczur kot lew zyrafa papuga tygrys goryl kaczka bocian

karty G2: piotrek szczur goryl tygrys lew papuga zyrafa kot kaczka bocian

STARCIE 5

- wyciągnięto: piotrek

karty G1: szczur kot lew zyrafa papuga tygrys goryl kaczka bocian piotrek

karty G2: szczur goryl tygrys lew papuga zyrafa kot kaczka bocian

STARCIE 6

- wyciągnięto: szczur

karty G1: kot lew zyrafa papuga tygrys goryl kaczka bocian piotrek

karty G2: goryl tygrys lew papuga zyrafa kot kaczka bocian

STARCIE 7

- wyciągnięto: goryl

karty G1: kot lew zyrafa papuga tygrys kaczka bocian piotrek

karty G2: tygrys lew papuga zyrafa kot kaczka bocian

STARCIE 8

- wyciagnieto: kot

karty G1: lew zyrafa papuga tygrys kaczka bocian piotrek

karty G2: tygrys lew papuga zyrafa kaczka bocian

STARCIE 9

- wyciagnieto: tygrys

karty G1: lew zyrafa papuga kaczka bocian piotrek

karty G2: lew papuga zyrafa kaczka bocian

STARCIE 10

- wyciagnieto: lew

karty G1: zyrafa papuga kaczka bocian piotrek

karty G2: papuga zyrafa kaczka bocian

STARCIE 11

- wyciagnieto: papuga

karty G1: zyrafa kaczka bocian piotrek

karty G2: zyrafa kaczka bocian

STARCIE 12

- wyciagnieto: zyrafa

karty G1: kaczka bocian piotrek

karty G2: kaczka bocian

STARCIE 13

- wyciagnieto: kaczka

karty G1: bocian piotrek

karty G2: bocian

STARCIE 14

- wyciagnieto: bocian

karty G1: piotrek

karty G2:

Wygrywa G2 !!!

D:\PPS_2021_22_P1\LAB_2021_22\project_lab_6a\Debug\project_lab_6a.exe (process 3688) exited with code 0.

Press any key to close this window . . .