

Zadania na I termin poprawkowy z PJC2

Na ocenę 3.0 - jedno zadanie:

1. Opracować program wczytujący ze standardowego wejścia słowa składające się tylko z liter i wyświetlający w osobnych wierszach na ekranie te słowa, które zawierają liczbę i rodzaj znaków podane w wierszu poleceń. Można założyć, że wszystkie wprowadzane dane będą poprawne. Przykładowo dla uruchomienia programu **\$ prog a 3** i danych na standardowym wejściu:

```
aaa bbb ccacasa dd   eeeeeaa sss ddd   4aa444 d1ddaaa bbaccadda
```

program powinien wyświetlić:

```
aaa
ccacasa
eeeeaaa
bbaccadda
```

2. Opracować program wczytujący ze standardowego wejścia ciągi znaków oddzielone spacją i wyświetlający te te ciągi, które zawierają zadany ciąg znaków (podawany w wierszu poleceń programu). Można założyć, że wszystkie wprowadzane dane będą poprawne.

Przykładowo dla wiersza poleceń **\$ prog abb** i znaków na standardowym wejściu:

```
ababbssh shhhh aa abbbb ssss dwkkw   aaa oeiufffl
abbbb aaa
ababssh sbbbb ssss dwkkw   aaa oeiufffl
aass abb ddd
```

program powinien wyświetlić:

```
ababbssh abbbb abbbb abb
```

3. Opracować program wczytujący z pliku wiersze i wyświetlający te, które zawierają zadaną liczbę grup liter danego typu. Można założyć, że wszystkie wprowadzane dane będą poprawne. Przykładowo dla zadanej grupy liter **abc** i liczby wynoszącej **3** oraz znaków w pliku:

```
abcababcabca
shshsk
wwwl llsd dj
abcb ca dabcbaa jss abc
klja hhkla hahh ;a; habchhs www hajhhj
```

program powinien wyświetlić:

```
abcababcabca
abcb ca dabcbaa jss abc
```

4. Opracować program wczytujący z wiersza poleceń nazwę pliku i parami liczby oddzielone spacją i wyświetlający na ekranie wiersze z tego pliku o podanych numerach, jeśli liczba znaków w tym wierszu jest równa drugiej liczbie pary. Można założyć, że wszystkie wprowadzane dane będą poprawne. Przykładowo dla wiersza poleceń: **\$ prog plik_we.txt 1 2 2 5 3 10 5 1** oraz zawartości pliku plik_we.txt postaci:

```
aa
bb
ccdddeef
dd
e
faaf
```

program powinien wyświetlić:

```
aa
ccdddeef
e
```

5. Opracować program wczytujący z pliku dane tekstowe i zapisujący do drugiego pliku tylko liczby zawierając cyfry z zakresu podanego w wierszu poleceń podczas uruchomienia programu. Można założyć, że dane będą poprawne. Przykładowo uruchomienie programu:

```
$ prog 138 plik_we.txt plik_wy.txt
```

oraz zawartości pliku plik_we.txt:

```
2233
44
222
112123
4997
556
777
8
```

spowoduje zapisanie w plik_wy.txt tylko liczb zawierających cyfry 1, 3 i 8:

```
2233
112123
8
```

6. Opracować program wczytujący ze standardowego wejścia znaki i zapisujący do pliku wiersze, jeśli ich długość będzie z zakresu długości podanej w trakcie uruchomienia programu. Można założyć, że wprowadzane dane będą poprawne. Przykładowo uruchomienie programu: **\$ prog 10-20 plik_wy.txt** spowoduje zapisanie w plik_wy.txt wierszy o długości co najmniej 10 znaków i nie dłuższe niż 20 znaków.
7. Opracować program wczytujący ze standardowego wejścia znaki aż do napotkania 3 znaków końca wiersza z rzędu - markera końca wprowadzanych znaków. Po wykryciu tego markera program powinien wyświetlać podsumowanie zawierające liczbę cyfr danego typu, jakie znalazły się w strumieniu. Można założyć, że wszystkie dane będą poprawne.
8. Opracować program wczytujący ze standardowego wejścia liczby oddzielone spacją (do napotkania znaku końca wiersza) i zapisujący do zadanego pliku tekstowe ciągi znaków, będących zwiokrotnieniem ciągu przekazanego w wierszu poleceń, przy czym liczba wierszy pliku musi odpowiadać ilości liczb wprowadzonych przez standardowe wejście. Można założyć, że wszystkie dane będą poprawne. Przykładowo dla uruchomienia programu **\$ prog aabbcc** i znaków na wejściu:

```
1 2 1 3 2 1 1
```

program powinien zapisać w pliku tekstowym wiersze:

```
aabbcc
aabbccaabbcc
aabbcc
aabbccaabbccaabbcc
aabbccaabbcc
aabbcc
aabbcc
```

9. Opracować program wczytujący z pliku tekstowego wiersze i zapisujący do drugiego pliku te wiersze, które zawierają tylko cyfry a ich długość nie jest większa niż wartość podana jako argument uruchomienia programu. Można założyć, że wszystkie dane będą poprawne, a podawana długość nie będzie większa niż 128.