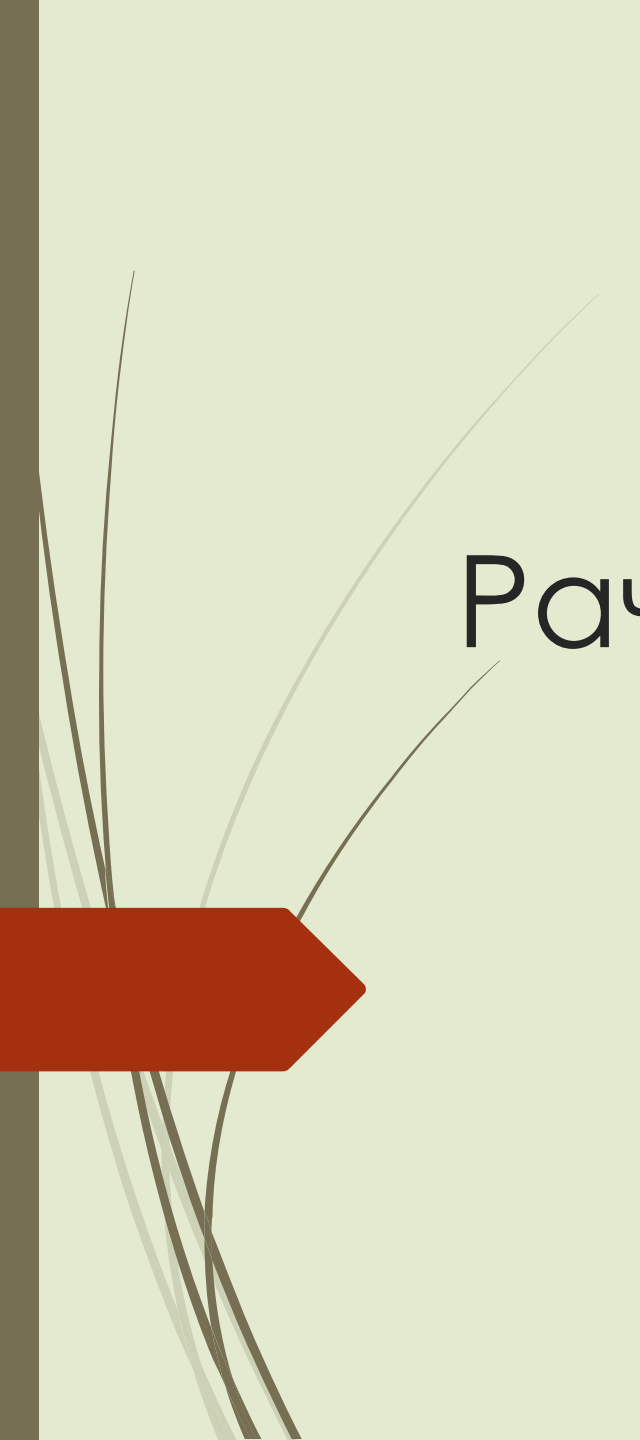




Рачунарство и информатика

Јелица Васиљевић

Одељење II см

Школска 2016/2017



Скупови - Увод

- (Подсећање) Типови података у Pascal-у
 - Прости типови
 - Стандардни (integer, real, char ...)
 - Нестандардни
 - Набројиви – boja = (crvena, zelena, zuta)
 - Интервални – ocena = 1..5
 - Сложени
 - Низови, скупови, слогови, датотеке и показивачи

Скупови - Увод

- Математички појам скупа : Више елемената који имају неко заједничко својство
- Појам скупа у Pascal – у : Више елемената који су **истог типа**
- Дефинисање скупа
 - Секција **type**

type

```
skupovniTip = set of <prost tip>
```

- Секција **var**

var

```
promenljiva : skupovniTip;
```

```
promenljiva1 : set of <prost tip>
```

Скупови – Базни тип

- Дефинисањем променљиве као скупа омогућава се да променљива узме вредност **било ког подскупа** базног типа (типа од кога се гради) укључујући **празан скуп**.
 - **Питање:** Колико **различитих** вредности може да узме променљива типа скиуп ?
- **Ограничење** базног типа скупа :
 - **Базни тип** не може бити **real**
 - **Базни тип** не може бити **integer** (али може неки његов део)
 - **Базни тип** не може имати више од 256 различитих вредности
 - **Ординате** свих вредности базног типа морају да припадају интервалу 0..255

Скупови – Карактеристике и означавање

■ Карактеристике

- Редослед елемената **није битан**
- Нема директног приступа елементима
- Не могу се учитавати/исписивати коришћењем стандардних I/O команди

■ Означавање

- У математици { елементи }
- У Pascal-у [**елементи**]
 - $S = [1,2,3]$ // Skup S sadrži tri elementa : 1,2 i 3
- Питање
 - $P = [3,1,2]$, $S=P$? Зашто ?

Скупови – Операције

- Над скуповима се могу вршити следеће операције :
 - Унија $+$
 - Пресек $*$ - има већи приоритет у односу на $+$ и $-$
 - Разлика $-$
 - Поређење
 - Једнакост $=$
 - Неједнакост $\neq$
 - Подскуп $\subseteq, \supseteq$
 - Испитивање припадности - **IN**

Скупови – Једноставна употреба

- Скраћено записивање више услова

```
bez skupova : if (a=10) or (a=15) or (a=20) then ...
```

```
sa skupovima : if a in [15,10,20] then ...
```



Скупови - основни пример

- Написати програм који читава два стринга и исписује све карактере који се појављују у првом стрингу а не појављују у другом. Карактере исписати у абecedном поретку
- 

Скупови - основни пример

```
program skupZad1;
type
  slova='a'..'z';
  znakovi = set of slova;
var
  znakoviPrvog : znakovi;
  znakoviDrugog : znakovi;
  skup : znakovi;
  prvi : string;
  drugi: string;
  i:integer;
  s:slova;

begin

  readln(prvi);
  readln(drugi);
  znakoviPrvog := [];
  znakoviDrugog := [];
```

```
for i:=1 to length(prvi) do
begin
  znakoviPrvog := znakoviPrvog + [ prvi[i]];
end;

for i:=1 to length(drugi) do
begin
  znakoviDrugog := znakoviDrugog + [ drugi[i]];
end;

skup := [];
for s:='a' to 'z' do
begin
  if ((s in znakoviPrvog) and not(s in znakoviDrugog)) then
skup:= skup + [s];
  if ((s in znakoviDrugog) and not(s in znakoviPrvog)) then
skup:= skup + [s];
  end;

  for s:= 'a' to 'z' do
    if ( s in skup) then writeln(s);

  readln;

end.
```