CS & IT ENGINEERING

Programming in C

Doubts

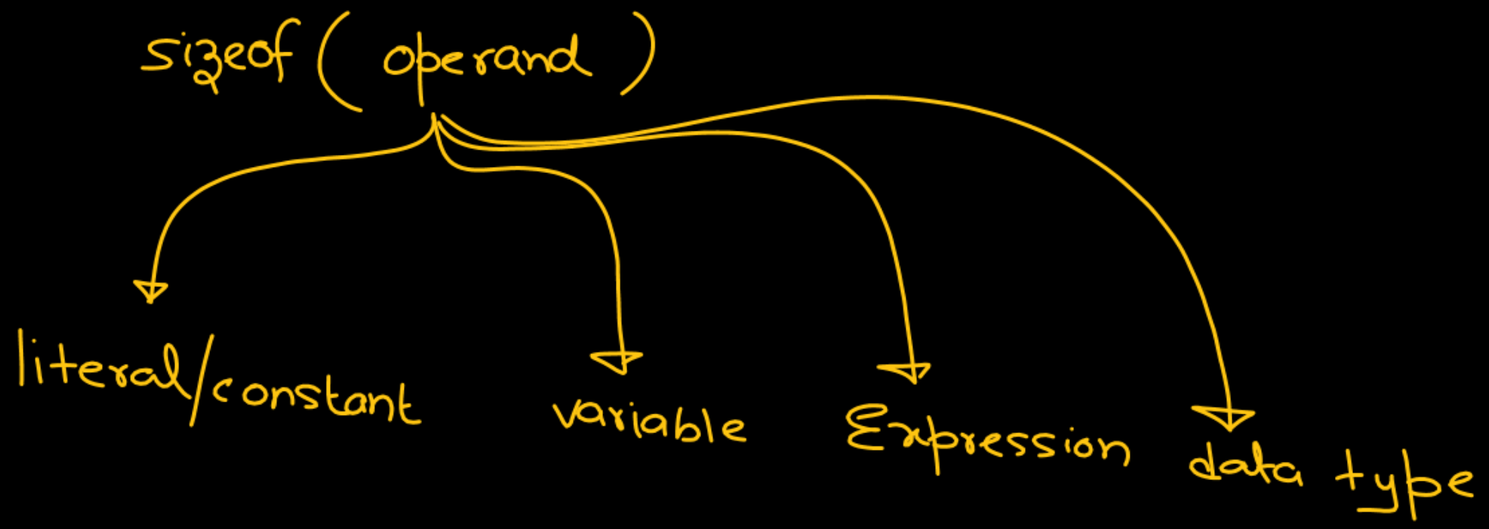
(In One Shot)

By- Pankaj Sharma sir

Problem Solving

sizeof operator

* unary operator : 1 operand
* return type : unsigned int
* Compile time


sizeof (operand)
literal/constant
variable
Expression
data type

int - 4
char - 1
float - 4
Address → 2 byte

int i = 3;

integer literal

```
printf("%u", sizeof(5));      4
printf("%u", sizeof(i));      4
printf("%u", sizeof(i+3));    4
printf("%u", sizeof(int));    4
```

int + int ⇒ int

char constant

```
pf("%d", sizeof('A'));      4
pf("%d", sizeof(10+1));     4
pf("%d", sizeof(12.38));
```

```
char ch = 'A';
pf("%d", sizeof(ch));
```

4

```
pf("%d", sizeof('A'+10));
```

* floating point constant/literal are treated as double value

```
char *p;
float *q;
int *r;
bf("%d",sizeof(p));
bf("%d",sizeof(q));
bf("%d",sizeof(r));
```

Same

```
bf("%d",sizeof(*p));
bf("%d",sizeof(*q));
bf("%d",sizeof(*r));
```

int **p, ***q;

"%u" "%d"

sizeof(p)
sizeof(*p)
Address

sizeof(**p)

sizeof(q)
sizeof(*q)
sizeof(**q)
Address

size of integer

sizeof(***q)

P is a pointer to a 2-D array of integer

```
int (*p)[2][3];
```

sizeof(p) → Address ✓

sizeof(*p)

H.W

sizeof(**p)

int (*p)[2][3]

int (*p)[2][3]

2×3 integer

6 integer

24 byte

*p ⇒ int [2][3]

**p ⇒ int [3]

⇒ 12 byte

sizeof(*p[0])

sizeof(*p[1])

1.

```
int i = 2;
printf("%d", sizeof(++i)); 4
printf("%d", i); 2
```

compile time

Eval

4242

2.

```
int i = 2;
pf("%d", sizeof(i++)); 4
pf("%d", i);           2
```

#—

3.

compile time

```
printf("%d", sizeof(printf("Hello")));
```

run time

```
pf("%d", sizeof(int));
```

O/P : 4

```
#define SQ(x) x*x
#include<stdio.h>
void main(){
    int i;
    i = SQ(5);
    pf("%d",i);
}
```

Pre Processing

```
void main(){
    int i;
    i = 5*5;
    pf("%d",i);
}
```

```
#define SQ(x) x*x
#include<stdio.h>
void main(){
    int i;
    i = SQ(5+2);
    pf("%d",i);
}
```

Pre Processing →

```
void main(){
    int i;
    i = 5+2*5+2;
    pf("%d",i);
}
```

17

data type & operator

short notes

10 Pages

(i) Cyclic property

(ii) Operator → Pri, Asso } && || short-ckt eval

Control flow:

(i) if-else : a) if(condition/expression)

b) else must be immediately after if scope end.

(ii) a) for(exp1/ϵ ; exp2/ϵ ; exp3/ϵ)

{

==

}

b) exp1, exp2, exp3 ⇒ optional

c) scope → first semicolon (by default)

① क्या पढ़ना है – imp

② क्या नहीं पढ़ना है – Most imp.

Review: C Programming & feedback


THANK YOU GW
SOLDIERS !