

Sample AgreeYa Solutions Technical Questions, Answers & Explanations

Q1. What is the output of the following program?

```
main()
{
int c[ ]={2.8,3.4,4,6.7,5};
int j,*p=c,*q=c;
for(j=0;j<5;j++) {
printf( %d ,*c);
++q; }
for(j=0;j<5;j++){
printf( %d ,*p);
++p; }
}
```

ANS: 2 2 2 2 2 2 3 4 6 5

Explanation:

Initially pointer c is assigned to both p and q. In the first loop, since only q is incremented and not c, the value 2 will be printed 5 times. In second loop p itself is incremented. So the values 2 3 4 6 5 will be printed.

Q2. What is the output of the following program?

```
main()
{
char *p;
printf("%d %d , sizeof(*p), sizeof(p));
}
```

Sample AgreeYa Solutions Technical Questions, Answers & Explanations

ANS: 1 2

Explanation:

*The sizeof() operator gives the number of bytes taken by its operand. P is a character pointer, which needs one byte for storing its value (a character). Hence sizeof(*p) gives a value of 1. Since it needs two bytes to store the address of the character pointer sizeof(p) gives 2.*

Q3. What is the output of the following program?

```
main()
{
extern int i;
i=20;
printf("%d, i);
}
```

ANS: Linker Error : Undefined symbol _i

Explanation:

extern storage class in the following declaration, extern int i; specifies to the compiler that the memory for i is allocated in some other program and that address will be given to the current program at the time of linking. But linker finds that no other variable of name i is available in any other program with memory space allocated for it. Hence a linker error has occurred.

Sample AgreeYa Solutions Technical Questions, Answers & Explanations

Q4. What is the output of the following program?

```
main()
{
printf(%x, -1<<4);
}
```

ANS: fff0

Explanation:

-1 is internally represented as all 1s. When left shifted four times the least significant 4 bits are filled with 0s. The %x format specifier specifies that the integer value be printed as a hexadecimal value.

Q5. What is the output of the following program?

```
void main()
{
int const * p=5;
printf(%d, ++(*p));
}
```

ANS: Compiler error: Cannot modify a constant value.

Explanation:

p is a pointer to a constant integer. But we tried to change the value of the constant integer.

Sample AgreeYa Solutions Technical Questions, Answers & Explanations

Q6. What is the output of the following program?

```
main()
float me = 1.1;
double you = 1.1;
if(me == you)
printf(I love U);
else
printf(I hate U);
}
```

ANS: I hate U

Explanation:

For floating point numbers (float, double, long double) the values cannot be predicted exactly. Depending on the number of bytes, the precision with of the value represented varies. Float takes 4 bytes and long double takes 10 bytes. So float stores 0.9 with less precision than long double.

Rule of Thumb:

Never compare or at-least be cautious when using floating point numbers with relational operators (== , >, <, <=, >=, !=) .

Q7. What is the output of the following program?

```
main()
{
int i=3;
```

Sample AgreeYa Solutions Technical Questions, Answers & Explanations

```
switch(i)
{
default:printf(zero);
case 1: printf(one);
break;
case 2:printf(two);
break;
case 3: printf(three);
break;
}
}
```

ANS: three

Explanation:

The default case can be placed anywhere inside the loop. It is executed only when all other cases does not match.

Q8. What is the output of the following program?

```
main()
{
char s[ ]=man;
int i;
for(i=0; s[ i ]; i++)
printf( %c%c%c%c, s[ i ], *(s+i), *(i+s), i[s]);
}
```

ANS: mmmm aaaa nnnn

Sample AgreeYa Solutions Technical Questions, Answers & Explanations

Explanation:

*s[i], *(i+s), *(s+i), i[s] are all different ways of expressing the same idea. Generally array name is the base address for that array. Here s is the base address. i is the index number/displacement from the base address. So, indirecting it with * is same as s[i].i[s] may be surprising. But in the case of C it is same as s[i].*

Q9. What is the output of the following program?

```
main()
{
static int var = 5;
printf("%d , var--);
if(var)
main();
}
```

ANS: 5 4 3 2 1

Explanation:

When static storage class is given, it is initialized once. The change in the value of a static variable is retained even between the function calls. Main is also treated like any other ordinary function, which can be called recursively.

Q10. What is the output of the following program?

Sample AgreeYa Solutions Technical Questions, Answers & Explanations

```
main()
{
int i=-1,j=-1,k=0,l=2,m;
m=i++&& j++&& k++ || l++;
printf("%d %d %d %d %d", i, j, k, l, m);
}
```

ANS: 0 0 1 3 1

Explanation:

Logical operations always give a result of 1 or 0. And also the logical AND (&&) operator has higher priority over the logical OR (||) operator. So the expression $i++\&\& j++\&\& k++$ is executed first. The result of this expression is 0 ($-1\&\&-1\&\&0=0$). Now the expression is $0\ ||\ 2$ which evaluates to 1 (because OR operator always gives 1 except for $0\ ||\ 0$ combination- for which it gives 0). So the value of m is 1. The values of other variables are also incremented by 1.