C PROGRAMMING 2021 SOLVE PAST PAPER

evaluate the expression 50+6-8/4*2%15-3

1. `50 + 6 - 8 / 4 * 2 % 15 - 3`
2. `50 + 6 - (8 / 4) * 2 % 15 - 3`
3. `50 + 6 - 2 * 2 % 15 - 3`
4. `50 + 6 - 4 % 15 - 3`
5. `50 + 6 - (4 % 15) - 3`
6. `50 + 6 - 4 - 3`
7. `56 - 4 - 3`
8. `52 - 3`
9. `49`

Therefore, the result of the expression `50 + 6 - 8 / 4 * 2 % 15 - 3` is `49`.

write the name of two escape sequences and their uses

Two commonly used escape sequences in C are:

1. **`\n`** - Newline Escape Sequence:
   - Use: It represents a newline character and is used to move the cursor to the beginning of the next line.
2. **`\t`** - Tab Escape Sequence:
   - Use: It represents a horizontal tab character and is used to insert a tab space.

These escape sequences are helpful for formatting output and controlling the positioning of text within the console.

what do you mean by loader

A loader is a program or component of an operating system that loads executable files into memory for execution, resolving any dependencies and preparing the program for execution by the processor.

OR

A loader is responsible for loading executable files into memory, ensuring that all required resources are available, and preparing them for execution by the computer's processor.

int 2a =6; is valid or not give reason

The statement `int 2a = 6;` is not valid in C because variable names in C must follow certain naming conventions:

1. Variable names must begin with a letter (a-z, A-Z) or an underscore (_).
2. After the first character, variable names may contain letters, digits (0-9), or underscores.
3. Variable names cannot be a keyword or a reserved word in C.

In this case, `2a` starts with a digit, which is not allowed as the first character of a variable name. Therefore, `int 2a = 6;` is not a valid declaration in C.

```
int a = 9, b = 3, c = -4;
if ((a + c) < 10 && (c < a - 3))
printf("%d\n", b + 3);
else
 printf("%d\n", a - 2); }
```

what is output

10

what is the scop and life time of static variable

Scope: A static variable declared within a function or block has a scope limited to that function or block. It is not accessible outside of the function or block where it is declared.
Lifetime: A static variable has a longer lifetime compared to automatic (local) variables. It retains its value between different function calls and is initialized only once.

easy:
A static variable in C has a limited scope within the block or function where it is declared. It retains its value between different invocations of that block or function. The static variable exists for the entire duration of the program, maintaining its value throughout.

if ptr is a pointer to int, and i and j are variable of int. whether the following statement is true / false ? give reason whatever the case may be ;

```
 ptr = & ( i+j);
```

The statement `ptr = &(i + j);` is false.

In C, the expression `&(i + j)` takes the address of the result of the addition `i + j`. However, the variable `ptr` is declared as a pointer to an integer (`int*`), which means it can only hold the address of an integer variable.

The expression `&(i + j)` represents the address of a temporary value resulting from the addition of `i` and `j`. This temporary value does not have a valid memory address that can be stored in the `ptr` variable.

To make the statement true, you would need to declare `ptr` as a pointer to a pointer to an integer (`int**`), or use a different approach to achieve your desired functionality.

```
int i , sum =1 ;
for (i=8;i>=2;i=i-2)
sum-=i;
printf("%d\n",sum);
```

what is output?

-27