

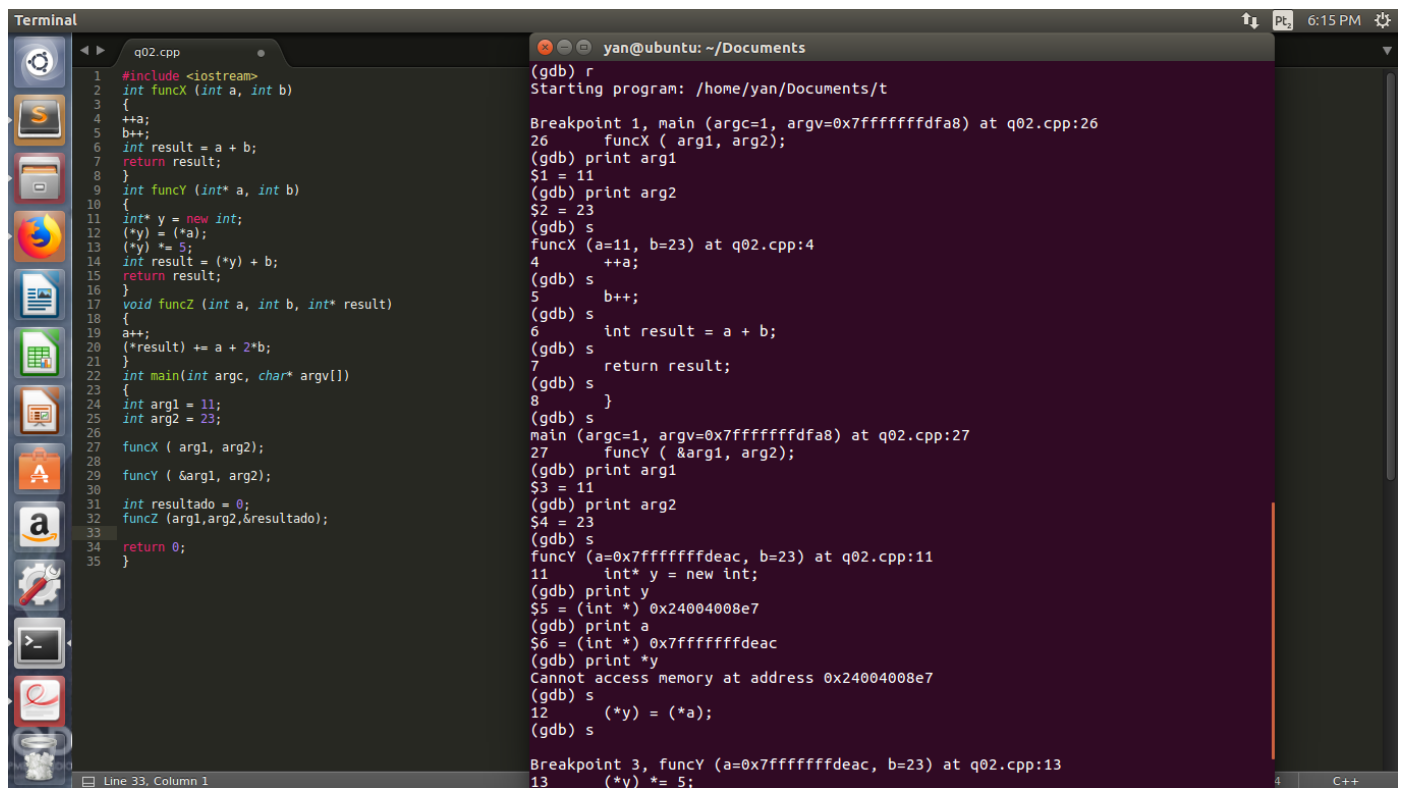
Questão 2: Para resolução desse problema foi adotada a tática de se colocar breakpoints antes da chamada de cada função e após o término da mesma, constantando através de métodos de print que o valor de arg1 e arg2 não se alteram, a seguir irei comentar cada função em separado do programa.

FuncX nessa função arg1 e arg2 são passados como simples parâmetros, não sendo feita referencição desse modo os valores não se alteram.

FuncY essa função gerou uma certa dúvida no primeiro momento, já que o arg1 é passado como parâmetro de referência, dentro da função sendo nomeado de 'a', após a atribuição feita de (\*y)=(\*a), imaginei que o valor de arg1 poderia sofrer alteração, mas ao dar print no valor da memória de 'y' e de 'a', constatei que os mesmos eram diferentes, assim chegando a conclusão que y só recebeu o valor de 'a', como na função só o 'y' é alterado o arg1 mantém o seu valor. Caso o 'a' sofresse alteração dentro da função o valor de arg1 seria alterado. Como arg2 não é passado por referência o valor continua o mesmo.

FuncZ na última função ocorre a mesma coisa que ocorreu na primeira, os dois valores não são referenciados logo só sofrem alteração no escopo da função, o valor de resultado sofre alteração mas ele não vem ao caso.

Conclusão geral: arg1 e arg2 não sofrem alteração, segue abaixo alguns prints do descrito



The screenshot shows a terminal window with two panes. The left pane displays the source code of a C++ program named q02.cpp. The code includes `<iostream>` and defines three functions: `funcX`, `funcY`, and `funcZ`, along with a `main` function. `funcX` increments a local variable `a` and returns it. `funcY` takes a reference to `arg1`, creates a new integer `y`, and assigns the value of `*a` to `*y`. `funcZ` increments a local variable `result` and returns it. The `main` function calls these functions with `arg1=11` and `arg2=23`. The right pane shows the GDB debugger's output, including breakpoints, variable values, and memory addresses, confirming that `arg1` and `arg2` remain unchanged throughout the program's execution.

```
Terminal
q02.cpp
1 #include <iostream>
2 int funcX (int a, int b)
3 {
4     ++a;
5     b++;
6     int result = a + b;
7     return result;
8 }
9 int funcY (int* a, int b)
10 {
11     int* y = new int;
12     (*y) = (*a);
13     (*y) *= 5;
14     int result = (*y) + b;
15     return result;
16 }
17 void funcZ (int a, int b, int* result)
18 {
19     a++;
20     (*result) += a + 2*b;
21 }
22 int main(int argc, char* argv[])
23 {
24     int arg1 = 11;
25     int arg2 = 23;
26
27     funcX ( arg1, arg2);
28
29     funcY ( &arg1, arg2);
30
31     int resultado = 0;
32     funcZ (arg1,arg2,&resultado);
33
34     return 0;
35 }

yan@ubuntu: ~/Documents
(gdb) r
Starting program: /home/yan/Documents/t

Breakpoint 1, main (argc=1, argv=0x7ffffffdfa8) at q02.cpp:26
26     funcX ( arg1, arg2);
(gdb) print arg1
$1 = 11
(gdb) print arg2
$2 = 23
(gdb) s
funcX (a=11, b=23) at q02.cpp:4
4     ++a;
(gdb) s
5     b++;
(gdb) s
6     int result = a + b;
(gdb) s
7     return result;
(gdb) s
8 }
(gdb) s
main (argc=1, argv=0x7ffffffdfa8) at q02.cpp:27
27     funcY ( &arg1, arg2);
(gdb) print arg1
$3 = 11
(gdb) print arg2
$4 = 23
(gdb) s
funcY (a=0x7ffffffdeac, b=23) at q02.cpp:11
11     int* y = new int;
(gdb) print y
$5 = (int *) 0x24004008e7
(gdb) print a
$6 = (int *) 0x7ffffffdeac
(gdb) print *y
Cannot access memory at address 0x24004008e7
(gdb) s
12     (*y) = (*a);
(gdb) s

Breakpoint 3, funcY (a=0x7ffffffdeac, b=23) at q02.cpp:13
13     (*y) *= 5;
```

```
Terminal
q02.cpp
1 #include <iostream>
2 int funcX (int a, int b)
3 {
4 ++a;
5 b++;
6 int result = a + b;
7 return result;
8 }
9 int funcY (int* a, int b)
10 {
11 int* y = new int;
12 (*y) = (*a);
13 (*y) *= 5;
14 int result = (*y) + b;
15 return result;
16 }
17 void funcZ (int a, int b, int* result)
18 {
19 a++;
20 (*result) += a + 2*b;
21 }
22 int main(int argc, char* argv[])
23 {
24 int arg1 = 11;
25 int arg2 = 23;
26 funcX ( arg1, arg2);
27 funcY ( &arg1, arg2);
28 int resultado = 0;
29 funcZ (arg1,arg2,&resultado);
30 return 0;
31 }

yan@ubuntu: ~/Documents
(gdb) print *y
Cannot access memory at address 0x24004008e7
(gdb) s
12 (*y) = (*a);
(gdb) s

Breakpoint 3, funcY (a=0x7fffffffdeac, b=23) at q02.cpp:13
13 (*y) *= 5;
(gdb) print y
$7 = (int *) 0x613c20
(gdb) print a
$8 = (int *) 0x7fffffffdeac
(gdb) s
14 int result = (*y) + b;
(gdb) print *y
$9 = 55
(gdb) print *a
$10 = 11
(gdb) s
15 return result;
(gdb) s
16 }
(gdb) s

Breakpoint 2, main (argc=1, argv=0x7fffffffdfa8) at q02.cpp:28
28 int resultado = 0;
(gdb) s
29 funcZ (arg1,arg2,&resultado);
(gdb) s
funcZ (a=11, b=23, result=0x7fffffffdeb0) at q02.cpp:19
19 a++;
(gdb) print arg1
No symbol "arg1" in current context.
(gdb) s
20 (*result) += a + 2*b;
(gdb) s
21 }
(gdb) s

Breakpoint 4, main (argc=1, argv=0x7fffffffdfa8) at q02.cpp:30
30 return 0;
(gdb) s
```