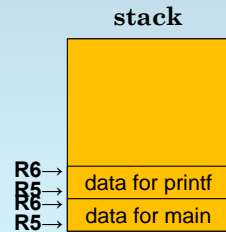


Example: main calls printf

main calls printf.

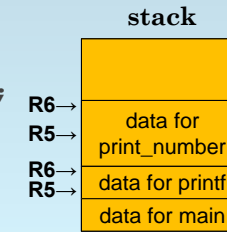
```
int main ()
{
  int32_t a = 42;
  printf ("%d", a);
  return 0;
}
```



Example: printf calls print_number

printf calls print_number.

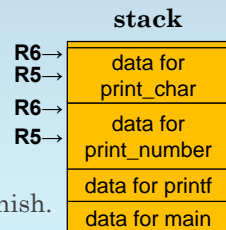
```
int main ()
{
  int32_t a = 42;
  printf ("%d", a);
  return 0;
}
```



Example: print_number calls print_char

print_number calls print_char.

```
int main ()
{
  int32_t a = 42;
  printf ("%d", a);
  return 0;
}
```



Eventually, all functions finish.

Stack and Heap Can Not Be Allowed to Collide

What happens if the heap and the stack collide?

- As discussed earlier,
- in LC-3/embedded ISA/inside OS, **silent data corruption**.
 - in programs with most ISAs, hardware detects and **crashes the program**.

