

# Lecture on Control Flow in Programs at Anonymous Institute

## Practice Worksheet

Instructor: Firstname Lastname, Anonymous Institute, Country  
firstname.lastname@institute.email

October 15, 2024

---

### Example Question and Answer

---

#### Program

```
0  a = read()
1  a = a + 1
2  # end
```

#### Structural abstraction

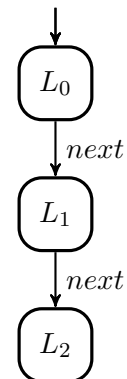
```
0  expression assignment
1  expression assignment
2  # end
```

---

#### Table of Control Transfer Functions

i	next	error
0	1	2
1	2	2
2		

#### Control Flow Graph



---

Execution path for the input  $a = 1$ .

$$L_0 \xrightarrow{\text{next}} L_1 \xrightarrow{\text{next}} L_2$$

## Section 1

---

**Program**

**Write the structural abstraction of the program.**

```
0  x = read()
1  y = 1
2  z = x + y
3  # end
```

---

**Fill in the table of Control Transfer Functions with the appropriate locations for the program.**

i	next	error
0		
1		
2		
3		

**Draw the Control Flow Graph (CFG) for the program.**

---

**Trace the actual execution of the program for the input  $x = 2$ .**

## Section 2

---

Program

Write the structural abstraction of the program.

```
0  x = read()
1  if x > 10:
2      y = 2
3  else:
4      y = x
5      x = x * 2
6  z = x - y
7  # end
```

---

Fill in the table of Control Transfer Functions with the appropriate locations for the program.

i	next	true	false	error
0				
1				
2				
3				
4				
5				
6				
7				

**Draw the Control Flow Graph (CFG) for the program.**

---

**Trace the actual execution of the program for the input  $x = 6$ .**

## Section 3

---

Program

Write the structural abstraction of the program.

```
0  i = read()
1  x = 5
2  while i > 0:
3      i = i - 1
4      y = x * i
5      continue
6  x = i + y
7  # end
```

---

Fill in the table of Control Transfer Functions with the appropriate locations for the program.

i	next	true	false	error
0				
1				
2				
3				
4				
5				
6				
7				

**Draw the Control Flow Graph (CFG) for the program.**

---

**Trace the actual execution of the program for the input  $i = 2$ .**

## Section 4

---

**Program**

**Write the structural abstraction of the program.**

```
0  def add(a, b):
1      c = a + b
2
3      return c
4
5  def modify(x):
6      x = x * 2
7
8      k = add(x, 1)
9
10     return k
11
12 m = add(5, 3)
13
14 n = modify(m)
15
16 # end
```

---

**Fill in the table of Control Transfer Functions with the appropriate locations for the program.**

i	next	call	ret	error
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				



**Draw the Control Flow Graph (CFG) for the program.**

---

**Trace the actual execution of the program.**

## Section 5

---

**Program**

**Write the structural abstraction of the program.**

```
0  def func(x):  
1      y = 5//x  
2      return x - 1  
3  i = read()  
4  while i >= 0:  
5      i = func(i)  
6      continue  
7  # end
```

---

**Fill in the table of Control Transfer Functions with the appropriate locations for the program.**

i	next	true	false	call	ret	error
0						
1						
2						
3						
4						
5						
6						
7						

**Draw the Control Flow Graph (CFG) for the program.**

---

**Trace the actual execution of the program for the input  $i = 1$ .**

## Space for Rough Work