

Ma. Corazon C. Macaraig
BSCS 3 - 2
2021-10699
DBTK

1. Simulate the expressions using your code (hand written).

Program 1:

$b = 3, c = 2, f = 1, i = 6$
 $m = 5 \% b * c - (20/10) + f * (20/10) + i$
play(m)

Simulation:

$b = 3$
 $c = 2$
 $f = 1$
 $i = 6$
 $m = 5 \% b * c - (20/10) + f * (20/10) + i$
 $m = 5 \% 3 * 2 - (20/10) + 1 * (20/10) + 6$
 $m = 5 \% 3 * 2 - 2 + 1 * 2 + 6$
 $m = \underline{2 * 2 - 2 + 1 * 2 + 6}$
 $m = \underline{4 - 2 + 1 * 2 + 6}$
 $m = \underline{4 - 2 + 2 + 6}$
 $m = \underline{2 + 2 + 6}$
 $m = \underline{4 + 6}$
 $m = \underline{10}$

Problem 2:

$d = 7, e = 3, f = 9$
 $c = 2 * (9 + 8 - d) / e + f$
play(c)

Simulation:

$d = 7$
 $e = 3$
 $f = 9$
 $c = 2 * (9 + 8 - d) / e + f$
 $c = 2 * (9 + 8 - 7) / 3 + 9$
 $c = 2 * (12 - 7) / 3 + 9$
 $c = \underline{2 * 5 / 3 + 9}$
 $c = \underline{10 / 3 + 9}$
 $c = \underline{3.33 + 9}$
 $c = \underline{12.33}$

Program 3:

$a = 5, b = 0, c = 10$
 $m = a * b + c - (1/4) + 7$
play(m)

Simulation:

$a = 5$
 $b = 0$
 $c = 10$
 $m = a * b + c - (1/4) + 7$
 $m = 5 * 0 + 10 - (1/4) + 7$
 $m = 5 * 0 + 10 - 0.25 + 7$
 $m = \underline{0 + 10 - 0.25 + 7}$
 $m = \underline{10 - 0.25 + 7}$
 $m = \underline{9.75 + 7}$
 $m = \underline{16.75}$

2. Run your compiler and display the output to validate your simulation.

Program 1:

```
Terminal  
10.0
```

Program 2:

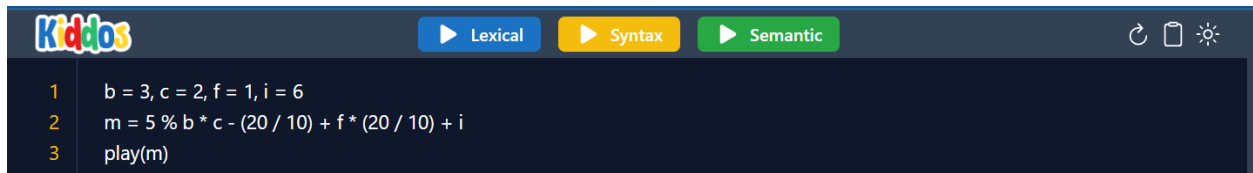
```
Terminal  
12.333333333333334
```

Program 3:

```
Terminal  
16.75
```

- Submit the screenshot of your code.

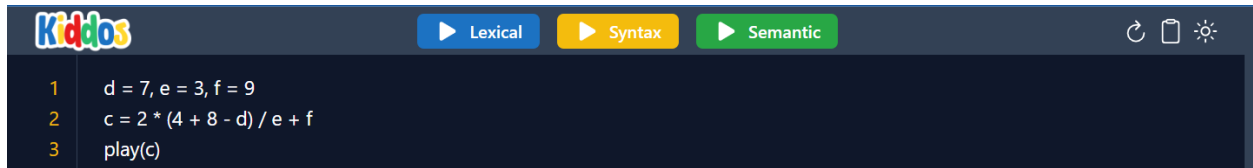
Program 1:



The screenshot shows the Kiddos IDE interface. At the top, there is a header with the 'Kiddos' logo and three buttons: 'Lexical' (blue), 'Syntax' (yellow), and 'Semantic' (green). To the right of these buttons are icons for refresh, copy, and settings. The main editor area has a dark background and contains three lines of code, each preceded by a line number (1, 2, 3) in yellow. The code is as follows:

```
1 b = 3, c = 2, f = 1, i = 6
2 m = 5 % b * c - (20 / 10) + f * (20 / 10) + i
3 play(m)
```

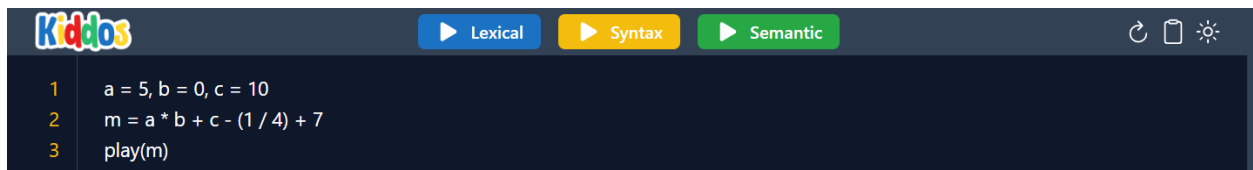
Program 2:



The screenshot shows the Kiddos IDE interface. At the top, there is a header with the 'Kiddos' logo and three buttons: 'Lexical' (blue), 'Syntax' (yellow), and 'Semantic' (green). To the right of these buttons are icons for refresh, copy, and settings. The main editor area has a dark background and contains three lines of code, each preceded by a line number (1, 2, 3) in yellow. The code is as follows:

```
1 d = 7, e = 3, f = 9
2 c = 2 * (4 + 8 - d) / e + f
3 play(c)
```

Program 3:

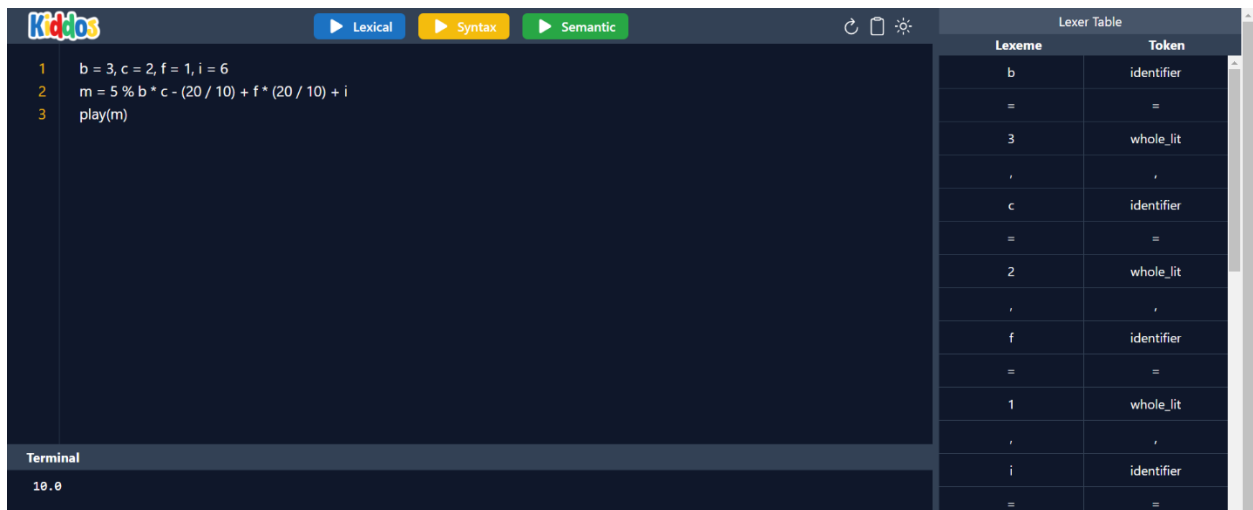


The screenshot shows the Kiddos IDE interface. At the top, there is a header with the 'Kiddos' logo and three buttons: 'Lexical' (blue), 'Syntax' (yellow), and 'Semantic' (green). To the right of these buttons are icons for refresh, copy, and settings. The main editor area has a dark background and contains three lines of code, each preceded by a line number (1, 2, 3) in yellow. The code is as follows:

```
1 a = 5, b = 0, c = 10
2 m = a * b + c - (1 / 4) + 7
3 play(m)
```

- Submit the screenshot of your output after running your compiler.

Program 1:



The screenshot shows the Kiddos IDE interface. At the top, there is a header with the 'Kiddos' logo and three buttons: 'Lexical' (blue), 'Syntax' (yellow), and 'Semantic' (green). To the right of these buttons are icons for refresh, copy, and settings. The main editor area has a dark background and contains three lines of code, each preceded by a line number (1, 2, 3) in yellow. The code is as follows:

```
1 b = 3, c = 2, f = 1, i = 6
2 m = 5 % b * c - (20 / 10) + f * (20 / 10) + i
3 play(m)
```

Below the editor area is a 'Terminal' section. It contains the text '10.0'. To the right of the editor and terminal is a 'Lexer Table' section. It contains a table with two columns: 'Lexeme' and 'Token'.

Lexeme	Token
b	identifier
=	=
3	whole_lit
,	,
c	identifier
=	=
2	whole_lit
,	,
f	identifier
=	=
1	whole_lit
,	,
i	identifier
=	=

Program 2:

Kiddos

▶ Lexical ▶ Syntax ▶ Semantic

```
1 d = 7, e = 3, f = 9
2 c = 2 * (4 + 8 - d) / e + f
3 play(c)
```

Terminal

12.333333333333334

Lexeme	Token
d	identifier
=	=
7	whole_lit
,	,
e	identifier
=	=
3	whole_lit
,	,
f	identifier
=	=
9	whole_lit
\n	newline
c	identifier
=	=

Program 3:

Kiddos

▶ Lexical ▶ Syntax ▶ Semantic

```
1 a = 5, b = 0, c = 10
2 m = a * b + c - (1 / 4) + 7
3 play(m)
```

Terminal

16.75

Lexeme	Token
a	identifier
=	=
5	whole_lit
,	,
b	identifier
=	=
0	whole_lit
,	,
c	identifier
=	=
10	whole_lit
\n	newline
m	identifier
=	=