Lembar Jawaban EDP
Nama : Anjany Risqiati

1. Menampilkan nilai maksimal/ terbanyak dari array binary (1, 0) yang berurutan. Bahasa pemrograman yang digunakan adalah Python.

   a. Source Code

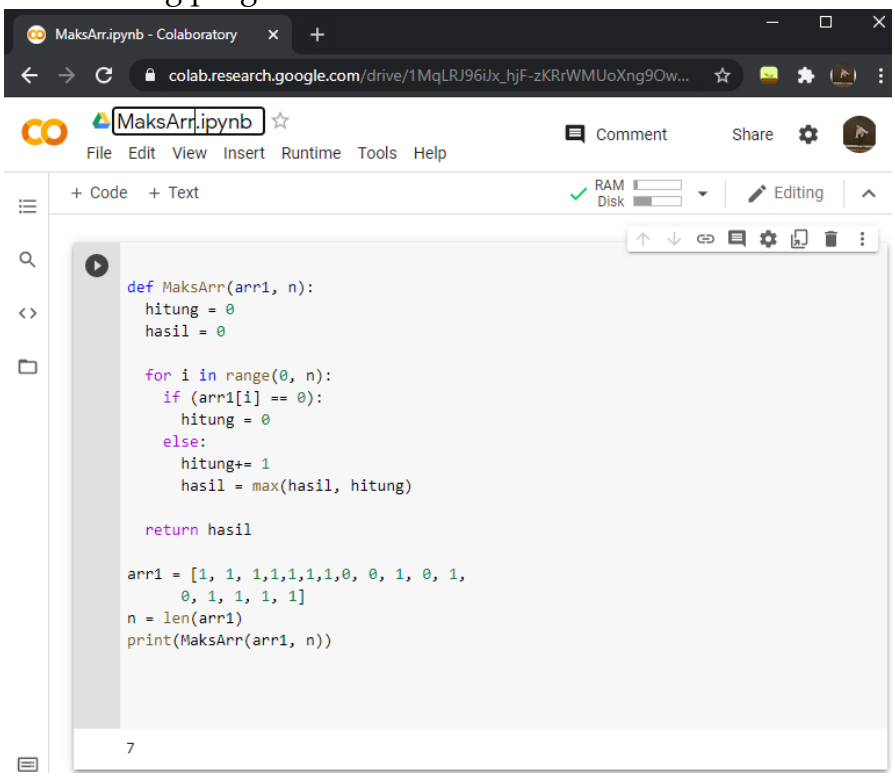
```
def MaksArr(arr1, n):
  hitung = 0
  hasil = 0

  for i in range(0, n):
    if (arr1[i] == 0):
      hitung = 0
    else:
      hitung+= 1
      hasil = max(hasil, hitung)

  return hasil

arr1 = [1, 1, 1,1,1,1,1,0, 0, 1, 0, 1,
     0, 1, 1, 1, 1]
n = len(arr1)
print(MaksArr(arr1, n))
```

   b. Running program dan hasil



2. Menampilkan string yang terbalik dengan fungsi rekursif. Bahasa pemrograman yang digunakan adalah Python.

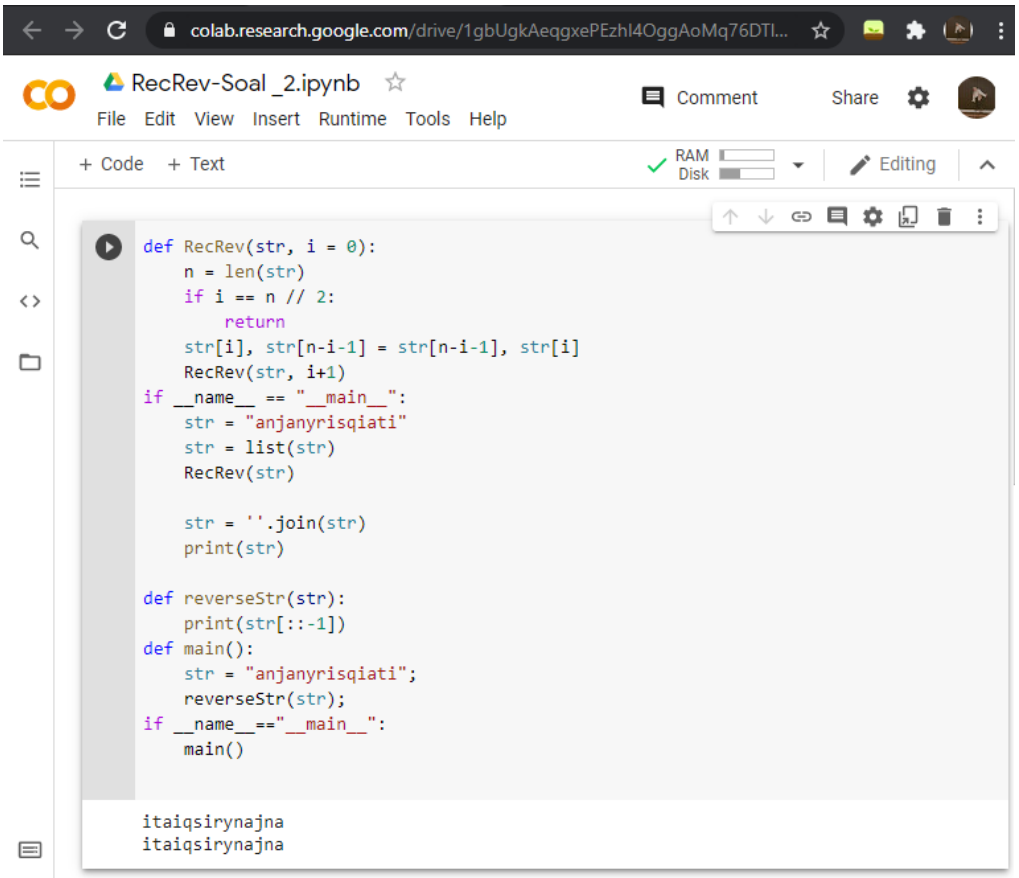
a. Source Code

```
def RecRev(str, i = 0):
    n = len(str)
    if i == n // 2:
        return
    str[i], str[n-i-1] = str[n-i-1], str[i]
    RecRev(str, i+1)
if __name__ == "__main__":
    str = "anjanyrisqiati"
    str = list(str)
    RecRev(str)

    str = ''.join(str)
    print(str)

def reverseStr(str):
    print(str[::-1])
def main():
    str = "anjanyrisqiati";
    reverseStr(str);
if __name__=="__main__":
    main()
```

b. Running program dan hasil



3. Membuat fungsi untuk melakukan pengecekan apakah ekspresi / bracket sesuai/ balance atau tidak. Bahasa pemrograman yang digunakan adalah Java.

a. Source Code

```
import java.util.*;

public class bracket1 {

        static boolean CekBracket(String expr)
        {

                Deque<Character> stack
                        = new ArrayDeque<Character>();

                for (int i = 0; i < expr.length(); i++)
                {

                        char x = expr.charAt(i);

                        if (x == '(' || x == '[' || x == '{')
                        {

                                stack.push(x);
                                continue;
                        }


                        if (stack.isEmpty())
                                return false;
                        char check;
                        switch (x) {
                        case ')':
                                check = stack.pop();
                                if (check == '{' || check == '[')
                                        return false;
                                break;

                        case '}':
                                check = stack.pop();
```

```
                        if (check == '(' || check == '[')
                                return false;
                        break;

                case ']':
                        check = stack.pop();
                        if (check == '(' || check == '{')
                                return false;
                        break;
                }
        }

        return (stack.isEmpty());
    }


    public static void main(String[] args)
    {
        String expr = "([{}])";

        // Function call
        if (CekBracket(expr))
            System.out.println("Bracket sudah seimbang ");
        else
            System.out.println("Bracket tidak seimbang ");
    }
}
```

b. Running program dan hasil

bracket1.java

```java
import java.util.*;

public class bracket1 {

    static boolean CekBracket(String expr)
    {

        Deque<Character> stack
            = new ArrayDeque<Character>();

        for (int i = 0; i < expr.length(); i++)
        {

            char x = expr.charAt(i);

            if (x == '(' || x == '[' || x == '{')
            {

                stack.push(x);
                continue;
            }


            if (stack.isEmpty())
                return false;
            char check;
            switch (x) {
            case ')':
                check = stack.pop();
                if (check == '{' || check == '[')
                    return false;
                break;

            case '}':
                check = stack.pop();
                if (check == '(' || check == '[')
                    return false;
                break;

            case ']':
                check = stack.pop();
                if (check == '(' || check == '{')
                    return false;
                break;
            }
        }

        return (stack.isEmpty());
    }


    public static void main(String[] args)
    {
        String expr = "([{}])";

        // Function call
        if (CekBracket(expr))
            System.out.println("Bracket sudah seimbang ");
        else
            System.out.println("Bracket tidak seimbang ");
    }
}
```

c. Hasil


Command Prompt
Microsoft Windows [Version 10.0.18363.1440]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\USER>E:
E:\>javac bracket1.java
E:\>java bracket1
Bracket sudah seimbang
E:\>