|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Öğrenci No** |  |  | **Sınav Tarihi** | 27.01.2024 |
| **Ad - Soyad** |  |  | **Değerlendirme Sonucu** |  |
| **İmza** |  |  |
|  |

**SORULAR**

**S.1. (40 p) Aşağıda verilen algoritmaları ve C programlarını kendinizi bilgisayar yerine koyarak izleyin. Değişken değişimlerini ve ekran çıktısını yazın?**

|  |  |
| --- | --- |
| **Cevap (a):** | **Cevap (b):** |
| |  |  | | --- | --- | | x | y | | 11 | 12 | | 14 | 3 | | 6 | 5 | | -2 | 7 | | 1 | -2 | | -7 | 0 |   Ekran Çıktısı  Sonuç**: -7** | |  |  |  |  | | --- | --- | --- | --- | | inputNumber | finalResult | n | result | | 4 | 42 | 4 | 1 | |  |  | 3 | 6 | |  |  | 2 | 20 | |  |  | 1 | 42 |   Ekran çıktısı  Sonuç: 42 |
| **Cevap (c):** | **Cevap (d):** |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | i | a | b | resultX | resultY | | 1  2  3  4  5  6 | 5 | 5 | 0  3  4 | 0  6  14 |   Ekran Çıktısı  Sonuç X: 4  Sonuç Y: 24 | |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | A | B | C | numberN | i | result | v1 | v2 | v3 | | 5  10 | 10  5 | 0  5040 | 7 | 1  2  3  4  5  6  7 | 1  2  6  24  120  720  5040 | AA | BB | CC |   Ekran Çıktısı  Sonuç A=10, B=5, C=5040 |

**S.2. (30 p)**

#include <stdio.h>

int AsalCarpim(int n) {

int carpim=1;

for (int i=1;i<=n;i++)

{bool asalmi=true;

for (int j=2;j<i;j++)

{if (i%j==0)

{asalmi=false;

break;}

}

if (asalmi) carpim\*=i;

}

return carpim;

}

int main() {

int sayi;

printf("Bir sayi girin: ");

scanf("%d", &sayi);

int sonuc = AsalCarpim(sayi);

printf("Asal sayıların çarpımı: %d\n", sonuc);

return 0;

}

**S.3. (30 p)**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <ctype.h>

int isDigit(char c) {

return c >= '0' && c <= '9';

}

int isOperator(char c) {

return c == '+' || c == '-' || c == '\*' || c == '/';

}

int charToInt(char c) {

return c - '0';

}

int applyOperation(int a, int b, char op) {

switch(op) {

case '+': return a + b;

case '-': return a - b;

case '\*': return a \* b;

case '/': return a / b;

}

return 0;

}

int evaluateTPN(char\* expr) {

int a = 0, b = 0;

char op;

char token = expr[0];

int i = 0;

while (token != NULL) {

if (i==0 && !isdigit(token)) return 0;

else if (i==1 && !isdigit(token)) return 0;

else if (i>=2)

{ if (i%2==0 && !isOperator(token)) return 0;

else if (i%2==1 && !isdigit(token)) return 0;

}

i++;

token = expr[i];

}

return 1;

}

int main() {

char rpnExpr[] = "34/5\*";

if (evaluateTPN(rpnExpr)) printf("Doğru");

else printf("Yanlış");

return 0;

}