דף תרגילים 5 יובל רשמן

**Ex1:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#define STUDENTS 30

#define TESTS 5

int caculateAvg(int sum, int n) {

return sum / n;

}

int main() {

int studentIndex, classSum = 0 ,studentSum = 0, testIndex, grade;

char name[20];

for (studentIndex = 0; studentIndex < STUDENTS; studentIndex++)

{

printf("enter yout name: \n");

scanf(" %s", &name);

studentSum = 0;

for (testIndex = 1; testIndex <= TESTS; testIndex++)

{

printf("enter test number %d grade: \n", testIndex);

scanf("%d", &grade);

studentSum += grade;

}

studentSum = caculateAvg(studentSum, TESTS);

printf("student %s averege grade is: %d \n", name, studentSum);

classSum += studentSum;

}

printf("class averege is %d", caculateAvg(classSum, STUDENTS));

return 0;

}

**Ex2:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

void isPrimeNumber(int num) { //print number if he is prine number

int index;

for (index = 2; index < num; index++)

{

if (!(num % index)) { /\* check if we can devide the number in the index number \*/

return;

}

}

printf("%d \n", num);

}

void allSmallerPrimeNumbers(int n) { // print all prime numbers smaller from n

int currentNumebr;

for (currentNumebr = 2; currentNumebr < n; currentNumebr++)

{

isPrimeNumber(currentNumebr);

}

}

int main() {

int num;

printf("enter number: \n");

scanf(" %d", &num);

while (num)

{

if (num % 2 != 0) //check if num is ezogi

{

allSmallerPrimeNumbers(num);

}

printf("enter number: \n");

scanf(" %d", &num);

}

return 0;

}

**Ex3:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

int sumDigit(int num) { //caculete the sum of the number

int sum = 0;

while (num > 0) {

sum += num % 10;

num /= 10;

}

return sum;

}

int numDigit(int num) { //caculete how many digits the number has

int digitCounter = 0;

while (num > 0) {

digitCounter++;

num /= 10;

}

return digitCounter;

}

int main() {

int num;

printf("enter number: \n");

scanf(" %d", &num);

while (num > 0) // run when num is positive

{

printf("number %d, digit number: %d, digit sum: %d \n", num, numDigit(num), sumDigit(num));

printf("enter number: \n");

scanf(" %d", &num);

}

return 0;

}

**Ex4:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

int oneEvenNumber(int num) { // check if number is even

while (num > 0)

{

if (!(num % 2))

{

return 1; //Represent True

}

num /= 10;

}

return 0; //Represent false

}

int main() {

int num, i, evenCounter = 0;

for (i = 1; i <= 10; i++)

{

printf("enter number %d: \n", i);

scanf("%d", &num);

evenCounter += oneEvenNumber(num);

}

printf("there is %d numbers with even digit out of 10", evenCounter);

return 0;

}

**Ex5:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

void guessGame(void)

{

srand(time(NULL));

int playAgain = 1, tries, num1, num2, count, randomNumber;

while (playAgain)

{

randomNumber = rand() % 9000 + 1000;/\* generate numbre between 1000 – 9999 \*/

printf("%d \n", randomNumber); // tell us what is the random number

for (tries = 1; tries <= 10; tries++)

{

count = 0;

num2 = randomNumber;

printf ("This is your %d guess:\nguess a 4 Digit Number: \n", tries);

scanf ("%d", &num1);

while (num1 > 0)

{

if (num1 % 10 == num2 % 10)

{

count++;

}

num1 = (num1 / 10);

num2 = (num2 / 10);

}

if (count == 4)

{

printf("Congrats the number was: %d, and you made it in %d Guesses \n", randomNumber, tries);

break;

}

else if(tries != 10)

{

printf("You got %d correct digits \n", count);

}

else {

printf("Sorry the number was: %d\n", randomNumber);

}

}

printf("If you want to continue press 1, if not press 0 \n");

scanf("%d", &playAgain);

}

}

**Ex6:**

**לא ממש הבנתי את התרגיל**

**Ex7:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

int getQuestions(int difficultLevel, char sine) {

srand(time(NULL));

int firstDigit, secondDigit, questionsLeft = 20, correctCounter = 0, needToReplace;

double guess, resault;

for (; questionsLeft > 0; questionsLeft--)

{

switch (difficultLevel) { // check the difficulty Level

case 1:

firstDigit = rand() % 20 + 1; // in level one number are from 1 - 20

secondDigit = rand() % 20 + 1;

switch (sine) {

case '\*':

resault = firstDigit \* secondDigit;

break;

case '+':

resault = firstDigit + secondDigit;

break;

case '-':

if (firstDigit < secondDigit) {

needToReplace = firstDigit;

firstDigit = secondDigit;

secondDigit = needToReplace;

}

resault = firstDigit - secondDigit;

break;

case '/':

if (firstDigit % 2 != 0) { //always devide by 2 in this level

firstDigit++;

}

secondDigit = 2;

resault = firstDigit / secondDigit;

break;

}

break;

case 2:

firstDigit = rand() % 50 + 1;// in level two number are from 1 - 50

secondDigit = rand() % 50 + 1;

switch (sine) {

case '\*':

resault = firstDigit \* secondDigit;

break;

case '+':

resault = firstDigit + secondDigit;

break;

case '-':

resault = firstDigit - secondDigit;

break;

case '/':

for (secondDigit = 3; secondDigit < firstDigit; secondDigit++) /\* make sure that resualt will always

be decimal\*/

{

if (!(firstDigit % secondDigit))

{

break;

}

}

if (firstDigit % secondDigit)

{

secondDigit = 1;

}

resault = firstDigit / secondDigit;

break;

}

break;

case 3:

firstDigit = rand() % 100 + 1; //in this level numbers are from 1 - 100

secondDigit = rand() % 100 + 1;

switch (sine) {

case '\*':

resault = firstDigit \* secondDigit;

break;

case '+':

resault = firstDigit + secondDigit;

break;

case '-':

resault = firstDigit - secondDigit;

break;

case '/':

resault = (double)firstDigit / secondDigit;

resault \*= 1000; /\* remove all other digit after 3 digits after the dot \*/

resault = (int)resault;

resault /= 1000; // the answer will be 3 digit max after the dot

break;

}

break;

}

printf("qustion number %d: \n", 20 - questionsLeft + 1);

printf("caculete: %d %c %d \n", firstDigit, sine, secondDigit);

scanf("%lf", &guess);

if (guess == resault)

{

printf("correct\n");

correctCounter++;

}

else {

if (difficultLevel == 3)

{

printf("wrong, the answer is: %lf \n", resault);

}

else

{

printf("wrong, the answer is: %d \n", (int)resault);

}

}

}

return correctCounter;

}

int main() {

int wantToPlay = 1, difficulty;

char sine;

while (wantToPlay)

{

printf("lets start new game: \n");

printf("enter dificulty level (1 - 3): \n");

scanf(" %d", &difficulty);

printf("enter sine: \n");

scanf(" %c", &sine);

int reasult = getQuestions(difficulty, sine);

printf("you answer %d/20 correct \n", reasult);

printf("do you want to start a new game? (1 = yes 0 = no) \n");

scanf(" %d", &wantToPlay);

}

return 0;

}