Aufgabe 2

i)

#include <stdio.h>

#include <stdlib.h>

#include <limits.h>

int main(void)

{

int n;

double x;

n = rand();

n = - n;

x = (double)((n) / RAND\_MAX);

printf("%i\n%f\n", n, x);

return 0;

}

j)

#include <stdio.h>

#include <stdlib.h>

#include <ctype.h>

int main(void)

{

int n;

char c;

n = rand();

c = n % 128;

if(c > 47 && c < 58) {

printf("1\n");

}

else if(c > 64 && c < 91) {

printf("2\n");

} else if(islower((int)c)) {

printf("3\n");

} else if(isxdigit((int)c)) {

printf("4\n");

}

printf("%c\n", c);

printf("%i\n", c);

return 0;

}

Aufgabe 4

a)

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int sum(int w[], int size);

int main(void)

{

int i;

int size = 30;

int w[50];

srand(time(NULL));

for(i = 0; i < 50; ++i)

{

w[i] = rand();

}

printf("%i\n",sum(w, size));

}

int sum(int w[], int size)

{

int i;

int k = 0;

for(i = 0; i < size; ++i) {

k += w[i];

}

return k;

}

b)

#include <stdio.h>

int array\_to\_upper(char v[], char w[], int size);

int main(void)

{

int i;

char v[12] = {'I', 'n', 'n', 'f', 'o', 'r', 'm', 'a', 't', 'i', 'k', '1'};

char w[12];

int size = sizeof(v) / sizeof(v[0]);

if(array\_to\_upper(v, w, size)) {

for(i = 0; i < size; ++i) {

printf("%c\t", w[i]);

}

}

return 0;

}

int array\_to\_upper(char v[], char w[], int size)

{

int i, k;

for(i = 0; i < size; ++i)

{

if((v[i] > 64 && v[i] < 90) || (v[i] > 96 && v[i] < 122))

{

if(v[i] > 96 && v[i] < 122)

{

v[i] -= 32;

w[i] = v[i];

}

else

{

w[i] = v[i];

}

k = 1;

}

else

{

k = 0;

}

}

return k;

}

c)

#include <stdio.h>

void manipulate\_and\_print(int v[], int size);

int main(void)

{

int v[9] = {3, -3, 6, 13, 7, 8, 30, -10, 1028};

int size = sizeof(v) / sizeof(v[0]);

manipulate\_and\_print(v, size);

return 0;

}

void manipulate\_and\_print(int v[], int size)

{

int i;

for(i = 0; i < size; ++i) {

if(v[i] >= 0) {

switch(v[i] % 3) {

case 0:

printf("%i\n", v[i] / 2);

break;

case 1:

printf("%i\n", ++v[i]);

break;

case 2:

printf("%i\n", -v[i]-1);

break;

default:

printf("fehler");

}

} else

printf("\n");

}

}

d)

#include <stdio.h>

void copy\_and\_transform\_digits(char v[], char w[], int size);

int main(void)

{

int i;

char v[12] = {'0', '1', '2', '3', '4', '5', 'A', 'B', 'C', 'd', 'E', '9'};

char w[12];

int size = sizeof(v) / sizeof(v[0]);

printf("size: %i\n", size);

copy\_and\_transform\_digits(v, w, size);

printf("Feld v: ");

for(i = 0; i < size; ++i) {

printf("%c\t", v[i]);

}

printf("\n");

printf("Feld w: ");

for(i = 0; i < size; ++i) {

printf("%c\t", w[i]);

}

return 0;

}

void copy\_and\_transform\_digits(char v[], char w[], int size)

{

int i;

for(i = 0; i < size; ++i) {

if(v[i] > 47 && v[i] < 58) {

w[i] = 138 - v[i];

} else {

w[i] = v[i];

}

}

}