Übungen zu C++ Wintersemester 2014
Aufgabenblatt 11

Aufgabe 28
Erzeugen Sie eine Folge von 20 000 Bit und überprüfen Sie, ob diese Folge den Tests aus FIPS 140 genügt.

```
// Zu Aufgabe 28
#include <iostream>
#include <random>
using namespace std;

int main() {
   unsigned long ul = 1234567;
   mt19937 eng (ul);
   uniform_int_distribution<int> dist (0, 127);
   for (int i = 1; i < 201; ++i) {
         cout << dist(eng) << "   ";
      if  (i%10 == 0)
         cout << endl;
   }
   cout << endl;
}//main
```

Bemerkung: Sie können auch das random_device nutzen, das auf Intel-Rechnern eine Folge " echter"Zufallszahlen erzeugt.

```
/*
 19   127   116   70    69   98     96   125    96    13
 78    90    35   75    45   112     2     5    27    30
 47    83   126   18    71   100   103    22    35    33
 32    87    46   86    84   124    42    38    58   124
 12     0   104   112   15   114   105    50    22    76
 25    76    37   79    40   122    76   114    39    91
126    84    94   15   103   58     43    74    42     0
  1   125     7   53    39   91     16   104   117     3
 23    47    85   3     92   67     61    38     3    72
 22    58   123   74    23   108    33   104    67   111
 19    85    97   124   73   18      1    99    32   119
 22   109    46   105   42   109     2    81   113   112
 75    45    80    12   31   120    57    96    26    80
  2    25    14    36   33   14     11    33    26   127
 68   112     6   124   14   38     45   108    84    58
110    57    56    19   126   89   103    62    60    23
120   121    24    47   51   31     27    80   110    81
 22     6    30     5   50   46     42   118    70    89
  0    59   105    68   2    71     85    66    58   113
117    35    93    21   40   16     37    83     4    95
*/
```

Aus FIPS PUB 140-1 (11/01/1994)

Bemerkung: FIPS PUB 140-1 steht für Federal Information Processing Standards Publication 140-1.

Test für Generatoren von Zufallszahlen, Testobjekt ist jeweils eine Bitkette der Länge 20 000.

(i) Monobittest:

Der Test gilt als bestanden, falls

9 645 < Zahl der 1-Bit < 10 346

(ii) Pokertest:

Unterteile die Bitkette in 5000 nichtüberlappende 4-Bit Abschnitte. Bestimme die Häufigkeit f (i) $(0 \le i \le 15)$ für jedes 4-Bit-Musters; berechne χ^2 gemäß:

$$\chi^2 = \frac{16}{5000} * \left(\sum_{i=0}^{15} f(i)^2 \right) - 5000$$

Der Test gilt als bestanden, falls $1{,}03 < \chi^2 < 57{,}4$

(iii) Lauftest:

Sei B_i die Zahl der 1-Bit Läufe der Länge i (0<i<6),
sei B_6 die Zahl der 1-Bit Läufe der Länge > 5,
sei C_i die Zahl der 0-Bit Läufe der Länge i (0<i<6),
sei C_6 die Zahl der 0-Bit Läufe der Länge > 5.

Der Test gilt als bestanden, falls sämtliche B- und C-Werte innerhalb folgender Grenzen liegen.

Lauflänge	untere Grenze	obere Grenze
1	2267	2733
2	1079	1421
3	502	748
4	223	402
5	90	223
6	90	223

Ferner darf kein Lauf einer Länge größer als 33 existieren.

(iv) Wiederholungstest:

Produziert ein Zufallszahlengenerator Bitblöcke der Länge n, wobei n > 15 ist, dann dürfen nie zwei aufeinanderfolgende Bitblöcke gleich sein; ist die Produktionslänge l ≤ 15, dann sind für diesen Test jeweils n aufeinanderfolgend produzierte Bits zu einem Block der Länge n, wobei n > 15 zu wählen ist, zusammenzufassen.