

Mock eksamen 1

Github: <https://github.com/Skjalde4/Mock1-HelenaS-1997>

Azure: <https://portal.azure.com/#@edu.easj.dk/resource/subscriptions/f04f20df-7416-428c-9fcf-8f76406066df/resourcegroups/Mock1/providers/Microsoft.Web/sites/BilhusetHelenaS/appServices>

Opgave 2b:

På nedenstående billede kan man se et overblik over min code coverage.

Code Coverage Results					
helen_LAPTOP-I6K5TF0B 2019-10-23 14_14_					
Hierarchy	Not Covered (Blocks)	Not Covered (% Blocks)	Covered (Blocks)	Covered (% Blocks)	
helen_LAPTOP-I6K5TF0B 2019...	2	4,26 %	45	95,74 %	
skat.dll	0	0,00 %	18	100,00 %	
Skat	0	0,00 %	18	100,00 %	
Afgift	0	0,00 %	18	100,00 %	
BilAfgift(int)	0	0,00 %	9	100,00 %	
ElBilAfgift(int)	0	0,00 %	9	100,00 %	
testskat.dll	2	6,90 %	27	93,10 %	
TestSkat	2	6,90 %	27	93,10 %	
UnitTest1	2	6,90 %	27	93,10 %	
TestBilAfgiftNeg...	1	14,29 %	6	85,71 %	
TestBilAfgiftOve...	0	0,00 %	3	100,00 %	
TestBilAfgiftUnd...	0	0,00 %	3	100,00 %	
TestElbilAfgiftN...	1	14,29 %	6	85,71 %	
TestElbilAfgiftO...	0	0,00 %	3	100,00 %	
TestElbilAfgiftU...	0	0,00 %	3	100,00 %	
UnitTest1()	0	0,00 %	3	100,00 %	

Ud fra code coverage analysen kan jeg konkludere at min test har god code coverage, da mit procenttal ligger på 95,74% og de resterende 4,26% er tuborg klammer, som kan ses på nedenstående billede. Så derfor mener jeg at koden er testet fuldt igennem.

```
[TestMethod]
0 references
public void TestElbilAfgiftNegativ()
{
    int pris = -10000;

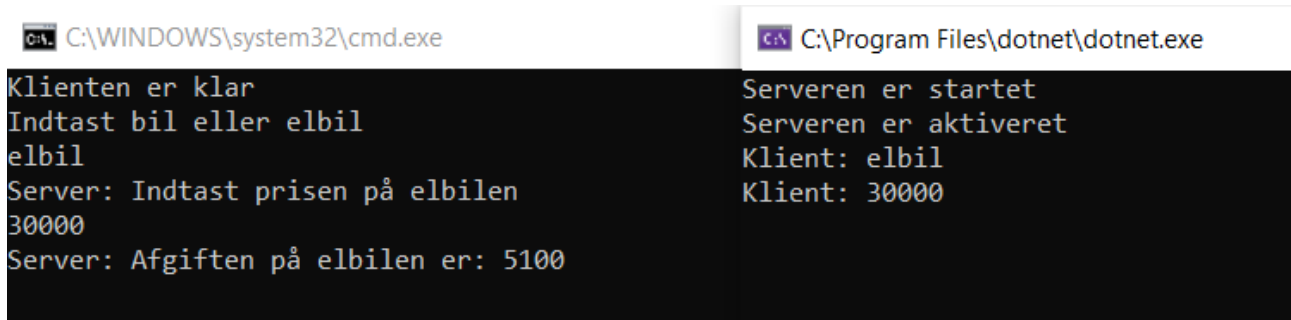
    try
    {
        afgift.ElBilAfgift(pris);
    }
    catch (Exception e)
    {
        Assert.AreEqual(expected: "Pris må ikke være mindre end 0", actual: e.Message);
    }
}
```

Opgave 3d:

Jeg implementerer en multi-trådet TCP server ved at benytte et while loop i main i serveren, som indeholder en task der afventer en klient. Jeg benytter factory til at lave serveren multi-trådet, da factory opretter et task objekt til hver klient der bliver opfanget.

```
Task.Factory.StartNew(() => afgift.DoIt());
```

Opgave 4c:



På ovenstående billede kan man tydeligt se at klienten er i kontakt med serveren. Man kan også her se at klienten søger afgiften på en elbil til 30000 kr. Dette kan man også læse sig frem til via WireShark, som man kan se på de to billeder nedenfor.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	127.0.0.1	127.0.0.1	TCP	46	51601 → 7000 [PSH, ACK] Seq=1 Ack=1 Win=10233 Len=2 [TCP segment of a reassembled PDU]
2	0.000047	127.0.0.1	127.0.0.1	TCP	44	7000 → 51601 [ACK] Seq=1 Ack=3 Win=10233 Len=0
3	0.000326	127.0.0.1	127.0.0.1	TCP	76	7000 → 51601 [PSH, ACK] Seq=1 Ack=3 Win=10233 Len=32 [TCP segment of a reassembled PDU]
4	0.000360	127.0.0.1	127.0.0.1	TCP	44	51601 → 7000 [ACK] Seq=3 Ack=33 Win=10233 Len=0
5	3.048940	127.0.0.1	127.0.0.1	TCP	51	51601 → 7000 [PSH, ACK] Seq=3 Ack=33 Win=10233 Len=7 [TCP segment of a reassembled PDU]
6	3.048984	127.0.0.1	127.0.0.1	TCP	44	7000 → 51601 [ACK] Seq=33 Ack=10 Win=10233 Len=0
7	3.049278	127.0.0.1	127.0.0.1	TCP	72	7000 → 51601 [PSH, ACK] Seq=33 Ack=10 Win=10233 Len=28 [TCP segment of a reassembled PDU]
8	3.049314	127.0.0.1	127.0.0.1	TCP	44	51601 → 7000 [ACK] Seq=10 Ack=61 Win=10233 Len=0
9	7.124369	127.0.0.1	127.0.0.1	TCP	51	51601 → 7000 [PSH, ACK] Seq=10 Ack=61 Win=10233 Len=7 [TCP segment of a reassembled PDU]
10	7.124415	127.0.0.1	127.0.0.1	TCP	44	7000 → 51601 [ACK] Seq=61 Ack=17 Win=10233 Len=0
11	7.124675	127.0.0.1	127.0.0.1	TCP	75	7000 → 51601 [PSH, ACK] Seq=61 Ack=17 Win=10233 Len=31 [TCP segment of a reassembled PDU]
12	7.124711	127.0.0.1	127.0.0.1	TCP	44	51601 → 7000 [ACK] Seq=17 Ack=92 Win=10233 Len=0
13	9.446996	127.0.0.1	239.255.255.250	SSDP	169	M-SEARCH * HTTP/1.1
14	9.449992	172.17.3.146	224.0.0.251	MDNS	71	Standard query 0x0000 ANY LAPTOP-IGKSTF0B.local, "QM" question
15	9.450384	fe80::11e6:f5ab:d7c...	ff02::fb	MDNS	91	Standard query 0x0000 ANY LAPTOP-IGKSTF0B.local, "QM" question
16	9.450700	172.17.3.146	224.0.0.251	MDNS	109	Standard query response 0x0000 AAAA fe80::11e6:f5ab:d7ca:fd72 A 172.17.3.146
17	9.450853	fe80::11e6:f5ab:d7c...	ff02::fb	MDNS	129	Standard query response 0x0000 AAAA fe80::11e6:f5ab:d7ca:fd72 A 172.17.3.146
18	9.451276	fe80::11e6:f5ab:d7c...	ff02::1:3	LLMNR	85	Standard query 0xd71e ANY LAPTOP-IGKSTF0B
19	9.451529	172.17.3.146	224.0.0.252	LLMNR	65	Standard query 0xd71e ANY LAPTOP-IGKSTF0B

> Frame 5: 51 bytes on wire (408 bits), 51 bytes captured (408 bits) on interface 0	
> Null/Loopback	
> Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1	
> Transmission Control Protocol, Src Port: 51601, Dst Port: 7000, Seq: 3, Ack: 33, Len: 7	

0000	02 00 00 00 45 00 00 2f	65 a2 40 00 80 06 00 00E.. / e. @
0010	7f 00 00 01 7f 00 00 01	c9 91 1b 58 f6 50 c0 1fX.P...
0020	64 35 49 21 50 18 27 f9	03 36 00 00 65 6c 62 69	d5!lP...6..elbi
0030	6c 0d 0a		1..

På billedet ovenfor kan man se at klienten som det først fortæller serveren at der er tale om en Elbil, på billedet nedenfor kan man se at klienten efterfølgende fortæller serveren at prisen på bilen er 30000 kr.

1 0.000000	127.0.0.1	127.0.0.1	TCP	40 51601 → 7000 [PSH, ACK] Seq=1 Ack=1 Win=10233 Len=2 [TCP segment of a reassembled PDU]
2 0.000047	127.0.0.1	127.0.0.1	TCP	44 7000 → 51601 [ACK] Seq=1 Ack=3 Win=10233 Len=0
3 0.000326	127.0.0.1	127.0.0.1	TCP	76 7000 → 51601 [PSH, ACK] Seq=1 Ack=3 Win=10233 Len=32 [TCP segment of a reassembled PDU]
4 0.000360	127.0.0.1	127.0.0.1	TCP	44 51601 → 7000 [ACK] Seq=3 Ack=33 Win=10233 Len=0
5 3.048940	127.0.0.1	127.0.0.1	TCP	51 51601 → 7000 [PSH, ACK] Seq=3 Ack=33 Win=10233 Len=7 [TCP segment of a reassembled PDU]
6 3.048984	127.0.0.1	127.0.0.1	TCP	44 7000 → 51601 [ACK] Seq=33 Ack=10 Win=10233 Len=0
7 3.049278	127.0.0.1	127.0.0.1	TCP	72 7000 → 51601 [PSH, ACK] Seq=33 Ack=10 Win=10233 Len=28 [TCP segment of a reassembled PDU]
8 3.049314	127.0.0.1	127.0.0.1	TCP	44 51601 → 7000 [ACK] Seq=10 Ack=61 Win=10233 Len=0
9 7.124369	127.0.0.1	127.0.0.1	TCP	51 51601 → 7000 [PSH, ACK] Seq=10 Ack=61 Win=10233 Len=7 [TCP segment of a reassembled PDU]
10 7.124415	127.0.0.1	127.0.0.1	TCP	44 7000 → 51601 [ACK] Seq=61 Ack=17 Win=10233 Len=0
11 7.124675	127.0.0.1	127.0.0.1	TCP	75 7000 → 51601 [PSH, ACK] Seq=61 Ack=17 Win=10233 Len=31 [TCP segment of a reassembled PDU]
12 7.124711	127.0.0.1	127.0.0.1	TCP	44 51601 → 7000 [ACK] Seq=17 Ack=92 Win=10233 Len=0
13 9.446996	127.0.0.1	239.255.255.250	SSDP	169 M-SEARCH * HTTP/1.1
14 9.449992	172.17.3.146	224.0.0.251	MDNS	71 Standard query 0x0000 ANY LAPTOP-I6K5TF08.local, "QM" question
15 9.450384	fe80::11e6:f5ab:d7c...	ff02::fb	MDNS	91 Standard query 0x0000 ANY LAPTOP-I6K5TF08.local, "QM" question
16 9.450700	172.17.3.146	224.0.0.251	MDNS	109 Standard query response 0x0000 AAAA fe80::11e6:f5ab:d7ca:fd72 A 172.17.3.146
17 9.450853	fe80::11e6:f5ab:d7c...	ff02::fb	MDNS	129 Standard query response 0x0000 AAAA fe80::11e6:f5ab:d7ca:fd72 A 172.17.3.146
18 9.451276	fe80::11e6:f5ab:d7c...	ff02::1:3	LLMNR	85 Standard query 0xd71e ANY LAPTOP-I6K5TF08
19 9.451529	172.17.3.146	224.0.0.252	LLMNR	65 Standard query 0xd71e ANY LAPTOP-I6K5TF08

Frame 9: 51 bytes on wire (408 bits), 51 bytes captured (408 bits) on interface 0

Null/Loopback

Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1

Transmission Control Protocol, Src Port: 51601, Dst Port: 7000, Seq: 10, Ack: 61, Len: 7

```
3000 02 00 00 00 45 00 00 2f 65 a6 40 00 80 06 00 00 ....E../e@....
3010 7f 00 00 01 7f 00 00 01 c9 91 1b 58 f6 50 c0 26 .....X.P.&
3020 64 35 49 3d 50 18 27 f9 a3 88 00 00 33 30 30 30 d5I=P.'.'...3000
3030 30 0d 0a 0..
```

På det sidste billede, som kan ses nedenfor, kan man her se serverens svar. Serveren informerer her kunden om hvad afgiften på bilen er, som i dette tilfælde er 5100 kr.

11 7.124675	127.0.0.1	127.0.0.1	TCP	75 7000 → 51601 [PSH, ACK] Seq=61 Ack=17 Win=10233 Len=31 [TCP segment of a reassembled PDU]
12 7.124711	127.0.0.1	127.0.0.1	TCP	44 51601 → 7000 [ACK] Seq=17 Ack=92 Win=10233 Len=0
13 9.446996	127.0.0.1	239.255.255.250	SSDP	169 M-SEARCH * HTTP/1.1
14 9.449992	172.17.3.146	224.0.0.251	MDNS	71 Standard query 0x0000 ANY LAPTOP-I6K5TF08.local, "QM" question
15 9.450384	fe80::11e6:f5ab:d7c...	ff02::fb	MDNS	91 Standard query 0x0000 ANY LAPTOP-I6K5TF08.local, "QM" question
16 9.450700	172.17.3.146	224.0.0.251	MDNS	109 Standard query response 0x0000 AAAA fe80::11e6:f5ab:d7ca:fd72 A 172.17.3.146
17 9.450853	fe80::11e6:f5ab:d7c...	ff02::fb	MDNS	129 Standard query response 0x0000 AAAA fe80::11e6:f5ab:d7ca:fd72 A 172.17.3.146
18 9.451276	fe80::11e6:f5ab:d7c...	ff02::1:3	LLMNR	85 Standard query 0xd71e ANY LAPTOP-I6K5TF08
19 9.451529	172.17.3.146	224.0.0.252	LLMNR	65 Standard query 0xd71e ANY LAPTOP-I6K5TF08

Frame 11: 75 bytes on wire (600 bits), 75 bytes captured (600 bits) on interface 0

Null/Loopback

Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1

Transmission Control Protocol, Src Port: 7000, Dst Port: 51601, Seq: 61, Ack: 17, Len: 31

```
1000 02 00 00 00 45 00 00 47 65 a8 40 00 80 06 00 00 ....E..G e@....
1010 7f 00 00 01 7f 00 00 01 1b 58 c9 91 64 35 49 3d .....X..d5I=
1020 f6 50 c0 2d 50 18 27 f9 6e 5a 00 00 41 66 67 69 .P.-P.'.' nZ..AfgI
1030 66 74 65 6e 20 70 c3 a5 20 65 6c 62 69 6c 65 6e ften p... elbilen
1040 20 65 72 3a 20 35 31 30 30 0d 0a er: 510 0..
```