Gagnvirk yfirferð verkefna og rafrænt prófakerfi

Hjalti Magnússon

YFIRFERÐ FORRITUNARVERKEFNA

- Tímafrek
- Ósamræmi
- Erfitt að finna villur
- Erfitt að gera sjálfvirkar prófanir

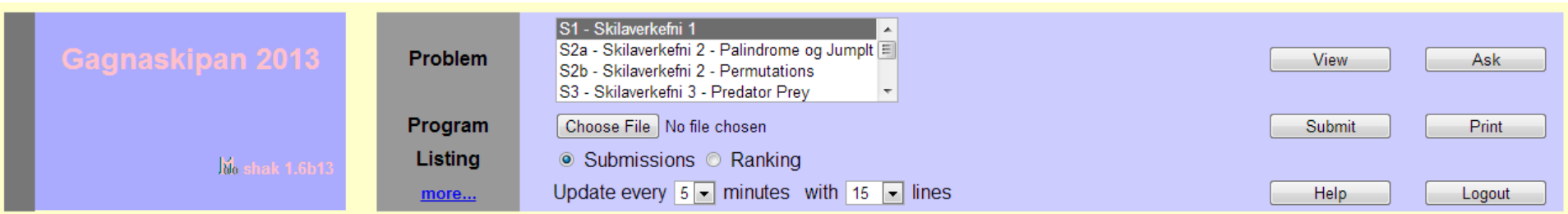
RAFRÆN YFIRFERÐ

- Mooshak
  - Yfirferðarþjónn
  - Upprunalega hannaður fyrir forritunarkeppnir
  - Hefur verið breytt til að nota til kennslu
- HR
  - Sérsniðum Mooshak að okkar þörfum
  - Skrifuðum ramma
    - Sérsniðnar prófanir
    - Sérsniðnar skýrslur um niðurstöður prófana

HVAÐ GERIR MOOSHAK?

- Kóðagreining
  - Stílreglur
  - Kóðagreinar (Cppcheck, Lint)
- Keyrslugreining
  - Minnisnotkun (Valgrind)
  - Keyrslutími
  - Prófanir
- Ef villa finnst við greiningu
  - Skýrsla

YFIRLIT



#	Absolute Time	Team	Problem	Language	Result
19600	23904:53:51	???????	S5	C++	0 Correct Output With Errors
19599	23904:52:57	???????	S5	C++	0 Accepted
19598	23904:48:31	???????	S5	C++	0 Correct Output With Errors
19597	23904:44:33	???????	S5	C++	0 Correct Output With Errors
19596	23904:37:55	???????	S5	C++	0 Correct Output With Errors
19595	23904:34:16	???????	S5	C++	0 Correct Output With Errors
19594	23904:29:55	???????	S5	C++	0 Wrong Answer
19593	23904:27:06	???????	S5	C++	0 Correct Output With Errors
19592	23904:26:10	???????	V11	C++	0 Static Analysis Error
19591	23904:25:21	???????	V11	C++	0 Compile Time Error
19590	23904:21:44	???????	S5	C++	0 Wrong Answer
19589	23904:21:24	???????	S5	C++	0 Accepted
19588	23904:19:29	???????	S5	C++	0 Accepted
19587	23904:18:33	???????	S5	C++	0 Static Analysis Error
19586	23904:15:57	???????	S5	C++	0 Static Analysis Error

Page 1 of 1307

KÓÐAGREINING

Static checker

N'SIQ Style Checker

```
jumpit.cpp:40:24 (A space around operator) Provide spaces after operator ','
jumpit.cpp:40:39 (A space around operator) Provide spaces after operator ','
jumpit.cpp:40:64 (A space around operator) Provide spaces after operator ','
jumpit.cpp:40:80 (A space around operator) Provide spaces after operator ','
jumpit.cpp:42:52 (A space around operator) Provide spaces after operator ','
jumpit.cpp:42:64 (A space around operator) Provide spaces b/w operator '+'
jumpit.cpp:42:66 (A space around operator) Provide spaces after operator ','
jumpit.cpp:46:52 (A space around operator) Provide spaces after operator ','
jumpit.cpp:46:67 (A space around operator) Provide spaces after operator ','
```

Vera++ Style Checker

```
jumpit.cpp:40: (T009) comma should be followed by whitespace
jumpit.cpp:40: (T009) comma should be followed by whitespace
jumpit.cpp:40: (T009) comma should be followed by whitespace
jumpit.cpp:40: (T009) comma should not be preceded by whitespace
jumpit.cpp:40: (T009) comma should be followed by whitespace
jumpit.cpp:42: (T009) comma should be followed by whitespace
jumpit.cpp:42: (T009) comma should be followed by whitespace
jumpit.cpp:46: (T009) comma should be followed by whitespace
jumpit.cpp:46: (T009) comma should be followed by whitespace
```

Static Analysis error

Valgrind Corrector

More information about the error messages produced by Valgrind can be found here.

```
==7677== Memcheck, a memory error detector
==7677== Copyright (C) 2002-2010, and GNU GPL'd, by Julian Seward et al.
==7677== Using Valgrind-3.6.0 and LibVEX; rerun with -h for copyright info
==7677== Command: ./a.out
==7677== Parent PID: 7676
==7677==
==7677==
==7677== HEAP SUMMARY:
==7677==     in use at exit: 32 bytes in 2 blocks
==7677==   total heap usage: 20 allocs, 18 frees, 344 bytes allocated
==7677==
==7677== 8 bytes in 1 blocks are indirectly lost in loss record 1 of 2
==7677==    at 0x4A0674C: operator new[](unsigned long) (vg_replace_malloc.c:305)
==7677==    by 0x400E3D: Permutations::insert(int, Node*, Node*&) (permutations.cpp:97)
==7677==    by 0x400FC2: Permutations::permutate(int*, int) (permutations.cpp:139)
==7677==    by 0x400FA5: Permutations::permutate(int*, int) (permutations.cpp:137)
==7677==    by 0x400C0C: Permutations::generate(int) (permutations.cpp:28)
==7677==    by 0x400ABC: main (main.cpp:9)
==7677==
==7677== 32 (24 direct, 8 indirect) bytes in 1 blocks are definitely lost in loss record 2 of 2
==7677==    at 0x4A06C8E: operator new(unsigned long) (vg_replace_malloc.c:261)
==7677==    by 0x400DEC: Permutations::insert(int, Node*, Node*&) (permutations.cpp:93)
==7677==    by 0x400FC2: Permutations::permutate(int*, int) (permutations.cpp:139)
==7677==    by 0x400FA5: Permutations::permutate(int*, int) (permutations.cpp:137)
==7677==    by 0x400C0C: Permutations::generate(int) (permutations.cpp:28)
==7677==    by 0x400ABC: main (main.cpp:9)
==7677==
==7677== LEAK SUMMARY:
==7677==    definitely lost: 24 bytes in 1 blocks
==7677==    indirectly lost: 8 bytes in 1 blocks
==7677==      possibly lost: 0 bytes in 0 blocks
==7677==    still reachable: 0 bytes in 0 blocks
==7677==         suppressed: 0 bytes in 0 blocks
==7677==
==7677== For counts of detected and suppressed errors, rerun with: -v
==7677== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 6 from 6)
```

INNTAK

Input

```
2

{}
{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32}

{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32}
{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32}
```

Difference between obtained and expected output

	Obtained output			Expected output	
f	1	1	f	1	1
	2	0		2	0
t	3	1	t	3	0
	4	1		4	0

RANGT SVAR

Difference between obtained and expected output

	Obtained output			Expected output	
f	1	original: a	f	1	original: a
	2	reverse: a		2	reverse: a
	3	sorted: a		3	sorted: a
	4	rotateLeft 0: a		4	rotateLeft 0: a
	5			5	
	6	original: ab		6	original: ab
t			t	7	reverse: ba
				8	sorted: ab
				9	rotateLeft 0: ab
				10	rotateLeft 1: ba
				11	
				12	original: abc
				13	reverse: cba
				14	sorted: abc
				15	rotateLeft 0: abc
				16	rotateLeft 1: bca
				17	rotateLeft 2: cab
				18	
				19	original: abcd
				20	reverse: dcba
				21	sorted: abcd
				22	rotateLeft 0: abcd
				23	rotateLeft 1: bcda
				24	rotateLeft 2: cdab
				25	rotateLeft 3: dabc
				26	
				27	original: abcde
				28	reverse: edcba
				29	sorted: abcde
				30	rotateLeft 0: abcde
				31	rotateLeft 1: bcdea
				32	rotateLeft 2: cdeab
				33	rotateLeft 3: deabc
				34	rotateLeft 4: eabcd

Difference between obtained and expected output

	Obtained output			Expected output	
f	1	1	f	1	1
	2	1		2	1
	3	1 2		3	1 2
	4	2 1		4	2 1
	5	1 3 2		5	1 3 2
n	6	2 1	n	6	2 3 1
	7	1 3 2 4		7	1 3 2 4
n	8	4 2 1	n	8	4 2 3 1
	9	1 5 3 2 4		9	1 5 3 2 4
n	10	4 2 1	n	10	4 2 3 5 1
	11	1 5 3 2 4 6		11	1 5 3 2 4 6
n	12	6 4 2 1	n	12	6 4 2 3 5 1
	13	1 7 5 3 2 4 6		13	1 7 5 3 2 4 6
n	14	6 4 2 1	n	14	6 4 2 3 5 7 1
	15	1 7 5 3 2 4 6 8		15	1 7 5 3 2 4 6 8
n	16	8 6 4 2 1	n	16	8 6 4 2 3 5 7 1
	17	1 9 7 5 3 2 4 6 8		17	1 9 7 5 3 2 4 6 8
n	18	8 6 4 2 1	n	18	8 6 4 2 3 5 7 9 1
	19	1 9 7 5 3 2 4 6 8 10		19	1 9 7 5 3 2 4 6 8 10
t	20	10 8 6 4 2 1	t	20	10 8 6 4 2 3 5 7 9 1
	21	9 7 5 3 2 4 6 8 10		21	9 7 5 3 2 4 6 8 10
	22	9 7 5 3 2 4 6 8		22	9 7 5 3 2 4 6 8
	23	7 5 3 2 4 6 8		23	7 5 3 2 4 6 8
	24	7 5 3 2 4 6		24	7 5 3 2 4 6
	25	5 3 2 4 6		25	5 3 2 4 6
	26	5 3 2 4		26	5 3 2 4
	27	3 2 4		27	3 2 4
	28	3 2		28	3 2
	29	2		29	2
	30	Empty list		30	Empty list

AF HVERJU MOOSHAK?

- Kerfið er mjög opið
- Grunnvirkni
  - Mooshak tekur við skrá
  - „Þýðing“
  - Kóðagreining
  - Keyrsla á prófunum og keyrslugreining
- Leyfir fjölbreytta notkun

FJÖLBREYTNI

- Við höfum notað Mooshak til að prófa
  - C++
  - Java
  - Python
  - SML
  - Prolog
  - Befunge
  - Regular expressions

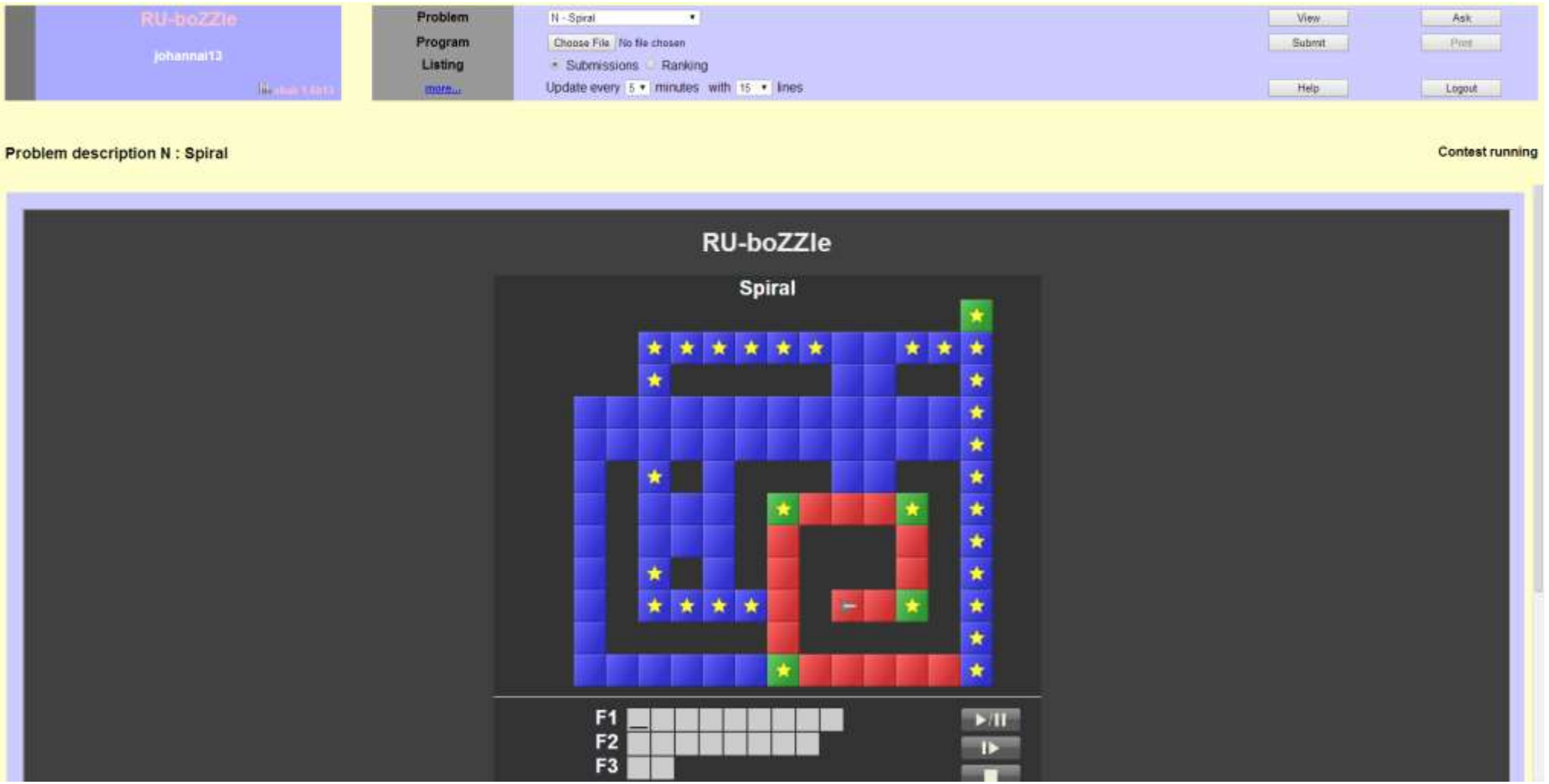
DULKÓÐUNARVERKEFNI

Test failed

Incorrect decryption

```
1: in cryptography, a Caesar cipher , also known as a Caesar 's cipher ,
                                     ^ Error here
```

ROBOZZLE


RU-boZZle
johannai13
Problem
Program
Listing
more...
N - Spiral
Choose File
No file chosen
Submissions
Ranking
Update every 5 minutes with 15 lines
View
Submit
Help
Ask
Print
Logout
Problem description N : Spiral
Contest running
RU-boZZle
Spiral
F1
F2
F3

LOKAPRÓF

- Lokapróf í forritun ennþá skrifleg!
- Margir ókostir
  - Nemendur óvanir að skrifa kóða á blað
  - Nemendur geta ekki keyrt og prófað kóða
  - Auðvelt að fá stig fyrir ranga lausn
  - Ónákvæmni í yfirferð

RAFRÆN LOKAPRÓF

- Kostir
  - Hægt að prófa lausn
  - Prófar raunverulega færni
  - Prófað í þekktu umhverfi
  - Erfiðara að fá stig fyrir ranga lausn
- Gallar
  - Ekki nógu einangruð
  - Erfitt að loka á net
  - Erfitt að loka á skráarkerfi
  - Þarfnast sérhæfðrar yfirsetu
  - Hægt að villa á sér heimildir
  - **Auðvelt að svindla**

NÝTT RAFRÆNT PRÓFAKERFI

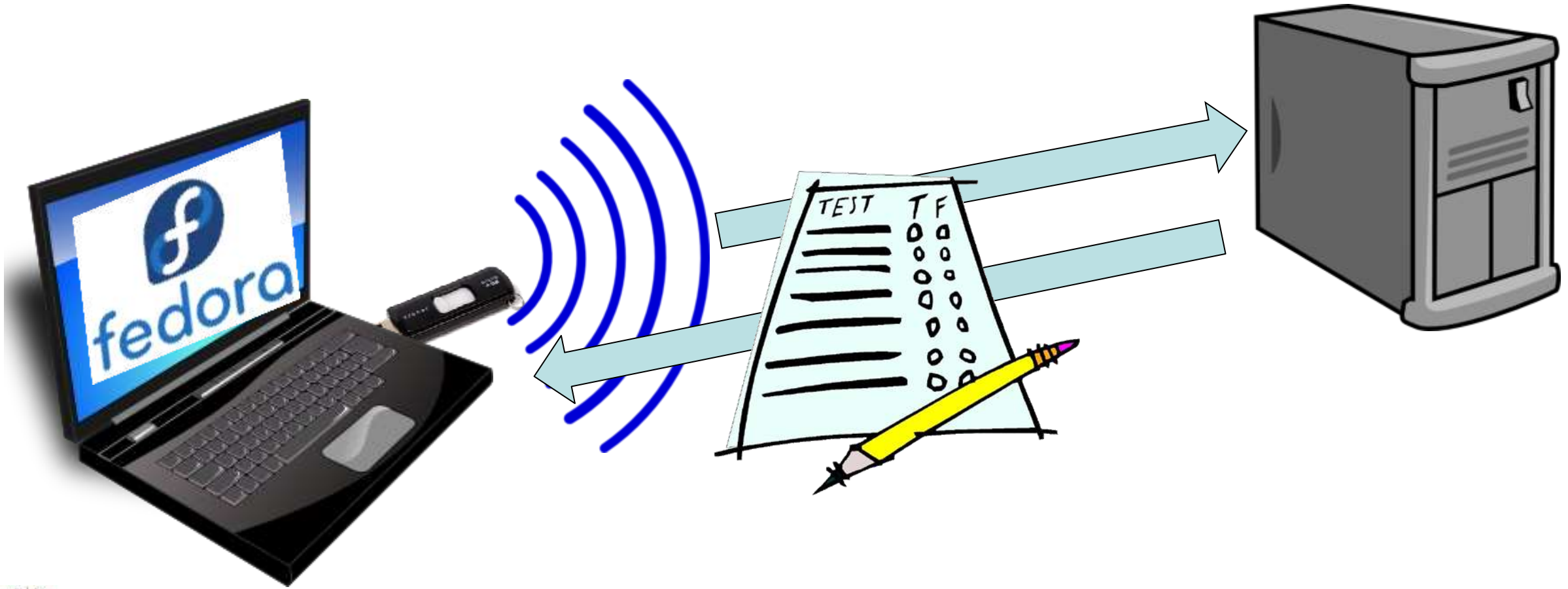
- Fullkomlega einangrað
- Algjör stjórn á netaðgangi
- Algjör stjórn á aðgangi gagna

PRÓFIÐ

PRÓFIÐ

- Linux ræst upp af USB-lykli
- Stýrikerfið keyrt í vinnsluminni
- Notandi hefur takmörkuð réttindi, getur t.d. ekki
  - tengst internetinu (eða öðrum vélum en þjóninum)
  - komist á harðan disk eigin tölvu
  - opnað aðra USB-lykla
  - kveikt á Bluetooth

FERLIÐ



ÞJÓNNINN

- Hefur ótakmörkuð réttindi að öllum vélum í prófi
- Afhendir nemanda próf
  - Setur prófið sjálft og fylgigögn á viðeigandi stað
- Passar að prófið opni á réttum tíma
- Tekur regluluega afrit af öllum vélum
  - Ef vél hrynur, þá má halda áfram þar sem frá var horfið
- Tekur við prófinu þegar nemandi skilar

HVAÐ MEÐ AÐ VILLA Á SÉR HEIMILDIR?

- Nemandi kallar á yfirsetu þegar hann lýkur prófi
- Nemandi og yfirseta fylla í sameiningu út hverju var skilað
- Báðir aðilar undirrita
- Staðfestir að nemandi skilaði prófi fyrir sjálfan sig

EKKI BARA FORRITUN

- Kerfið nýtist í fleiri tegundir prófa en forritunarpróf
- Fljótlegra að slá texta inn í tölvu
- Texti verður heildstæðari
- Auðvelt að setja upp formföst próf
  - Krossapróf
  - Eyðufyllingar
- Hægt er að stjórna nákvæmlega hvaða gögn nemandi hefur aðgang að

Takk fyrir