

NAME: _____

MATRIKEL

NUMMER: _____

Antestat 18-20

Aufgabe: Linear Page Table

Folgende Lineare Page Table (Page Größe: 1 KB) ist gegeben:

```
0x8000002a
0x00000000
0x00000000
0x00000000
0x8000003e
0x00000000
0x80000017
0x00000000
0x00000000
0x8000000e
0x00000000
0x00000000
0x8000001f
0x00000000
0x8000001a
0x80000001
```

Markieren Sie die richtigen Aussagen dazu:

- _____ Der virtuelle Adressraum ist 32 KB gross.
- _____ VPN 1 wurde bereits benutzt und zeigt auf Rahmen im Hauptspeicher.
- _____ Für die VPN werden 4 Bit benötigt.
- _____ VPN 9 hat das Valid Bit gesetzt.
- _____ Der Inhalt der Page Table weist auf einen Hauptspeicher hin, der grösser als 64 KByte ist.

Aufgabe: Speicherzugriffe

Folgende Page Table gegeben:

VPN	PFN
0	1
1	Not valid
2	3
3	Not valid

Die 1 Byte Page Table Einträge (PTE) hat. Der Adressraum eines Prozesses beträgt 128 Byte und die Größe einer Page ist 32 Byte. Der physikalische Speicher ist 128 Byte gross. Das Page Table Base Register ist auf die physikalische Adresse 16 gesetzt.

Betrachten wir nun folgende Befehlszeile, die von der virtuellen Adresse 70 ein Byte ins Register R1 lädt:

10: LOAD 70, R1

Die Instruction liegt an der virtuellen Adresse 10 im Adressraum des Prozesses.

Markieren Sie im (nächste Seite) dargestellten physikalischen Speicher:

- mit einem Rechteck die valide Virtuelle Seiten (Name/Label nicht vergessen!)
- mit einem Rechteck die Page Table (Name/Label nicht vergessen!)
- mit einem Kreis die Speicheradressen, die durch das Ausführen der Instruktion, inklusive dem Datenzugriff referenziert werden.
- Nummerieren Sie die mit einem Kreis markierten Adressen, so dass die Reihenfolge klar wird, in welcher die physikalischen Adressen referenziert werden.

Physical Memory:

0	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	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71
72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87
88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103
104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119
120	121	122	123	124	125	126	127

Aufgabe: TLB

Betrachten Sie folgendes Programm:

```
int i;  
int p[1024];  
for (i = 0; i < 1024; i++)  
    p[i] = 0;
```

Überlegen Sie sich das TLB Verhalten während der Loop Sequenz:

Frage: Wie viele Hits und viele Misses wird dieser Code durch die Loop auslösen, wenn er zum ersten Mal gestartet wird? Erklären Sie kurz Ihre Lösung und gehen Sie dazu von einer 1-KB PageSize und 4Byte Integer Größe aus.