

Constante de Kaprekar

A partir d'un nombre à 3 chiffres dont au moins un diffère des autres, le programme

- trie les 3 chiffres en décroissant pour obtenir un premier nombre abc,
- trie les 3 chiffres en croissant pour obtenir un deuxième nombre cba,

puis soustrait le deuxième nombre du premier et utilise le résultat comme nouveau départ ...etc...

jusqu'à **495** !

Pour le 6502 Microprocessor Kit :

Entrer le nombre en deux parties aux adresse **0001** (centaines) et **0002** (dizaines et unités)

exemple: [ADDR]0001[DATA]01[+]23[+] pour 123 (saisie en décimal)

Exécuter le programme :	[ADDR]0200[DATA][GO]
Vérifier le résultat :	[REG][\$01][REG][\$02] donne 198 (pour l'exemple 123)
Continuer le programme :	[DATA][GO] ou [ADDR]0297[DATA][GO]
Résultat suivant :	[REG][\$01][REG][\$02] donne 792 (pour l'exemple 123)
Continuer le programme :	[DATA][GO] ou [ADDR]0297[DATA][GO]
Résultat suivant :	[REG][\$01][REG][\$02] donne 693 (pour l'exemple 123)
Continuer le programme :	[DATA][GO] ou [ADDR]0297[DATA][GO]
Résultat suivant :	[REG][\$01][REG][\$02] donne 594 (pour l'exemple 123)
Continuer le programme :	[DATA][GO] ou [ADDR]0297[DATA][GO]
Résultat suivant :	[REG][\$01][REG][\$02] donne 495 (pour l'exemple 123)

ensuite continuer le programme donne 495 sans fin ! (constante de Kaprekar)

Le nombre de départ et les résultats sont en **décimal**.

TITLE **Kaprekar 495**
TITREPAGE **2** OF **2**
PAGE DE

0200	A5 01	DEB: LDA \$01	0233	A5 04	LDA \$04	026C	06 04	ASL \$04
0202	85 03	STA \$03	0235	E5 03	SBC \$03	026E	06 04	ASL \$04
0204	A5 02	LDA \$02	0237	B0 19	BCS PG3	0270	06 04	ASL \$04
0206	85 05	STA \$05	0239	4C 5D 02	JMP CAL	0272	18	CLC
0208	06 05	ASL \$05	023C	A6 04 PG1:	LDX \$04	0273	A5 04	LDA \$04
020A	06 05	ASL \$05	023E	A4 05	LDY \$05	0275	65 05	ADC \$05
020C	06 05	ASL \$05	0240	84 04	STY \$04	0277	85 04	STA \$04
020E	06 05	ASL \$05	0242	86 05	STX \$05	0279	06 07	ASL \$07
0210	46 05	LSR \$05	0244	4C 24 02	JMP TST	027B	06 07	ASL \$07
0212	46 05	LSR \$05	0247	A6 03 PG2:	LDX \$03	027D	06 07	ASL \$07
0214	46 05	LSR \$05	0249	A4 05	LDY \$05	027F	06 07	ASL \$07
0216	46 05	LSR \$05	024B	84 03	STY \$03	0281	18	CLC
0218	A5 02	LDA \$02	024D	86 05	STX \$05	0282	A5 07	LDA \$07
021A	85 04	STA \$04	024F	4C 24 02	JMP TST	0284	65 08	ADC \$08
021C	46 04	LSR \$04	0252	A6 03 PG3:	LDX \$03	0286	85 07	STA \$07
021E	46 04	LSR \$04	0254	A4 04	LDY \$04	0288	18	CLC
0220	46 04	LSR \$04	0256	84 03	STY \$03	0289	A5 03	LDA \$03
0222	46 04	LSR \$04	0258	86 04	STX \$04	028B	E5 06	SBC \$06
0224	18	TST: CLC	025A	4C 24 02	JMP TST	028D	85 01	STA \$01
0225	A5 05	LDA \$05	025D	F8	CAL: SED	028F	A5 04	LDA \$04
0227	E5 04	SBC \$04	025E	A5 05	LDA \$05	0291	E5 07	SBC \$07
0229	B0 11	BCS PG1	0260	85 06	STA \$06	0293	85 02	STA \$02
022B	18	CLC	0262	A5 04	LDA \$04	0295	00	BRK
022C	A5 05	LDA \$05	0264	85 07	STA \$07	0296	00	
022E	E5 03	SBC \$03	0266	A5 03	LDA \$03	0297	4C 00 02	JMP DEB
0230	B0 15	BCS PG2	0268	85 08	STA \$08			
0232	18	CLC	026A	06 04	ASL \$04			

Dump HEX :

```

0200: a5 01 85 03 a5 02 85 05 06 05 06 05 06 05 06 05
0210: 46 05 46 05 46 05 46 05 a5 02 85 04 46 04 46 04
0220: 46 04 46 04 18 a5 05 e5 04 b0 11 18 a5 05 e5 03
0230: b0 15 18 a5 04 e5 03 b0 19 4c 5d 02 a6 04 a4 05
0240: 84 04 86 05 4c 24 02 a6 03 a4 05 84 03 86 05 4c
0250: 24 02 a6 03 a4 04 84 03 86 04 4c 24 02 f8 a5 05
0260: 85 06 a5 04 85 07 a5 03 85 08 06 04 06 04 06 04
0270: 06 04 18 a5 04 65 05 85 04 06 07 06 07 06 07 06
0280: 07 18 a5 07 65 08 85 07 18 a5 03 e5 06 85 01 a5
0290: 04 e5 07 85 02 00 00 4c 00 02

```

Format Intel HEX :

```

:10020000a5018503a5028505060506050605060563
:100210004605460546054605a502850446044604EE
:100220004604460418a505e504b01118a505e50324
:10023000b01518a504e503b0194c5d02a604a40589
:10024000840486054c2402a603a405840386054c79
:100250002402a603a404840386044c2402f8a50502
:100260008506a5048507a50385080604060406047B
:10027000060418a504650585040607060706070693
:100280000718a5076508850718a503e5068501a5D4
:0A02900004e507850200004c00029F
:00000001FF

```