

**پروژه ۱ (BMI calculator):** برنامه‌ای با رابط کاربری گرافیکی (GUI) برای محاسبه شاخص توده بدنی (BMI)، که به صورت

$$BMI = \frac{\text{وزن بر حسب کیلوگرم}}{(\text{قد بر حسب متر})^2}$$

تعریف می‌شود، بنویسید. برنامه شما باید شامل جزئیات زیر باشد.

- دو فیلد برای وارد کردن اطلاعات کاربر، یکی برای وارد کردن وزن و دیگری برای وارد کردن قد
  - دارای قابلیت انتخاب واحد اعداد ورودی
    - وزن می‌تواند بر حسب کیلوگرم، گرم و پوند (۱ پوند = ۰,۴۵۳۵ کیلوگرم) وارد شود.
    - قد می‌تواند بر حسب متر، سانتی‌متر و اینچ (۱ اینچ = ۰,۰۲۵۴ متر) وارد شود.
  - توجه داشته باشید که برای محاسبه BMI، وزن باید به کیلوگرم و قد باید به متر تبدیل شوند.
  - خواندن اطلاعات وارد شده و سپس خالی کردن فیلدها برای این‌که برنامه آماده گرفتن ورودی‌های جدید شود.
  - محاسبه BMI و نمایش آن به کاربر.
- هرگونه شخصی‌سازی، بهبود و ارتقا، افزایش جزئیات کاربردی و دیگر کارهای خلاقانه و مفید در برنامه مجاز است و در صورت لزوم نمره تشویقی برای آن در نظر گرفته می‌شود.

**پروژه ۲ (آسانسور):** یک برج تازه تاسیس اقدام به استفاده از نوع خاصی از آسانسورها کرده که مدت زمان بالا رفتن و پایین رفتن آسانسور مساوی و به ازای هر طبقه یک ثانیه است. تیم برنامه‌نویسی مدار فرمان این آسانسور با مفروضات زیر، نیاز به همراهی شما برای ساخت این پروژه دارد.

۱- آسانسور در یک برج  $n$  طبقه و شامل طبقات ۱ تا  $n$  قرار دارد، ظرفیت آسانسور همواره از جمعیت ساکنین ساختمان بزرگتر است.

۲- در هر طبقه مانند  $q$  به نحوی که  $1 \leq q \leq n$ ، ممکن است فردی قصد بالا رفتن یا پایین رفتن داشته باشد، همچنین ممکن است در طبقات خاصی هیچ درخواستی نداشته باشیم.

۳- این آسانسور پیشرفته در بیرون از کابین دارای یک صفحه کلید است، به این معنی که قبل از سوار شدن افراد می‌توانند طبقه مقصد را به مدار فرمان اعلام کنند، در صورتی که قصد جابجایی نداشته باشند مدار فرمان ° دریافت می‌کند.

۴- همه درخواست‌های جابجایی در یک لحظه به مدار فرمان فرستاده می‌شود. برای مثال، چنین اطلاعاتی به شکل:

1:5

در قالب یک درخواست به مدار فرمان ارسال می‌شود. به این معنی که فرد ساکن طبقه ۱ قصد دارد به طبقه ۵ برود. یا مثلاً:

2:0

به این معنی است که فرد ساکن در طبقه ۲ قصد جابجایی ندارد. یا مثلاً:

3:1

به این معنی که فرد ساکن در طبقه ۳ قصد دارد به طبقه ۱ برود.

فرض کنید آسانسور در ابتدا در طبقه ۱ قرار دارد.

برنامه‌ای بنویسید که به ازای هر نوع جابجایی (بالا، پایین، بدون درخواست) از هر طبقه‌ای (۱ تا  $n$ ) با صرف کوتاه‌ترین زمان ممکن (چون بالا رفتن و پایین رفتن مدت یکسان طول می‌کشد، پس همان کوتاه‌ترین مسافت ممکن) همه افراد را به مقصد برسانیم.

این برنامه را باید با استفاده از ابزارهای oop و رابط کاربری گرافیکی (GUI) پیاده سازی کنید.

**پروژه ۳ (فضای برداری ۳-بعدی):** هدف این پروژه ایجاد یک class جدید در زبان Python برای معرفی بردارهای

۳-بعدی است. برای این منظور کلاس `vec_3` را به عنوان زیرکلاسی از کلاس `object` تعریف کنید. خواص (attributes) این کلاس عبارتند از:

۱- مؤلفه طولی: `length`

۲- مؤلفه عرضی: `width`

۳- مؤلفه عمودی: `height`

بعلاوه، این کلاس باید شامل توابع زیر باشد:

**Magic methods:**

۱- `__init__(self, a, y, z)`: برای ایجاد یک بردار سه بعدی با مؤلفه‌های  $x$ ،  $y$  و  $z$

۲- `__str__(self)`: برای نمایش یک بردار با مؤلفه‌های  $x$ ،  $y$  و  $z$  به صورت `[x; y; z]`

۳- `__add__(self, other)`: برای جمع دو بردار

۴- `__sub__(self, other)`: برای تفاضل دو بردار

۵- `__mul__(self, other)`: برای ضرب داخلی دو بردار

۶- `__eq__(self, other)`: برای بررسی تساوی دو بردار

۷- `__ne__(self, other)`: برای بررسی نامساوی بودن دو بردار

۸- `__abs__(self)`: برای محاسبه طول یک بردار

۹- `__ge__(self, other)`: برای بررسی بزرگتر یا مساوی بودن بردار `self` از بردار `other` بر اساس مقایسه طول دو بردار

۱۰- `__le__(self, other)`: برای بررسی کوچکتر یا مساوی بودن بردار `self` از بردار `other` بر اساس مقایسه طول دو بردار

۱۱- `__gt__(self, other)`: برای بررسی بزرگتر بودن بردار `self` از بردار `other` بر اساس مقایسه طول دو بردار

۱۲- `__lt__(self, other)`: برای بررسی کوچک‌تر بودن بردار `self` از بردار `other` بر اساس مقایسه طول دو بردار

### توابع دیگر:

۱- `unit(self)`: برای یک کردن بردار ناصفر. اگر بردار صفر باشد، ضمن نمایش یک پیغام خطا، خود بردار صفر را برمی‌گرداند.

۲- `out_prod(self, other)`: برای محاسبه ضرب خارجی دو بردار

۳- `dist(self, other)`: برای محاسبه فاصله دو بردار

۴- `angle(self, other)`: برای محاسبه زاویه بین دو بردار

۵- `proj(self, other)`: برای محاسبه تصویر بردار `self` بر روی بردار `other`

۶- `rej(self, other)`: برای محاسبه قرینه بردار `self` نسبت به بردار `other`

۷- `parall(self, other)`: برای بررسی موازی بودن دو بردار

۸- `prepend(self, other)`: برای بررسی عمود بودن دو بردار

۹- `triangle_area(self, other)`: برای محاسبه مساحت مثلث ایجاد شده بر روی دو بردار

۱۰- `parallelogram_area(self, other)`: برای محاسبه مساحت متوازی‌الاضلاع ایجاد شده بر

روی دو بردار

۱۱- `parallelepiped_vol(self, other1, other2)`: برای محاسبه حجم متوازی‌السطوح

ایجاد شده بر روی سه بردار

۱۲- `prism_vol(self, other1, other2)`: برای محاسبه حجم منشور ایجاد شده بر روی سه بردار

۱۳- `pyramid_vol(self, other1, other2)`: برای محاسبه حجم هرم ایجاد شده بر روی سه

برداری

در ادامه، یک زیر کلاس `vec_2` از کلاس `vec_3` ایجاد کنید، که ضمن به ارث بردن متودهای کلاس `vec_3`،

شامل تابع زیر نیز باشد:

`rotate(self, theta)`: برای دوران یک بردار به مرکز مبدا و اندازه زاویه `theta`

**پروژه ۴:** مطالعه و فهم کامل بخش ۴، ۱۰ کتاب Gutttag، پیاده‌سازی و آزمون پروژه مطرح شده در این بخش.



### تاریخچه:

انیگما یک ماشین رمزنگاری است که در پایان جنگ جهانی اول توسط یک شرکت آلمانی ساخته شد و از آن برای محافظت از اطلاعات تجاری و دیپلماتیک و نظامی استفاده می‌شد و سپس به طور گسترده در جنگ جهانی دوم مورد استفاده‌ی ارتش آلمان قرار گرفت (نوع نظامی آن پیشرفته‌ترین نوع انیگما بود). این ماشین بر مبنای یک الگوریتم ترکیبیاتی با کمک مکانیزم چرخ‌دنده‌های الکترومکانیکی کار می‌کند. در نهایت شکستن رمز آن که یکی از مهم‌ترین دلایل پیروزی متفقین بود حاصل تلاش‌های لهستانی‌ها و انگلیسی‌ها بود و از جمله معروف‌ترین آنها می‌توان به آقای آلن تورینگ اشاره کرد.

### برخی از مهم‌ترین نقطه قوت‌های این ماشین:

- اگر در plain text حروف تکراری موجود باشد، حروف تکراری در cipher text تولید نمی‌شود.
- یک تابع به تنهایی مسئولیت encryption و decryption را بر عهده دارد.
- تعداد حالتی که برای شکستن رمز آن باید امتحان کرد بسیار زیاد است و شکستن رمز آن با روش brute force حتی برای کامپیوترهای نسل جدید هم دشوار است.

### نحوه کارکرد دستگاه:

چرخ‌دنده‌ها وظیفه‌ی جابه‌جا کردن حروف را بر عهده دارند و هر حرف الفبا را به حرف دیگری متناظر می‌کنند (مثلاً اگر حرف B وارد آن شود حرف Z را خروجی دهد).  
سپس خروجی هر چرخ‌دنده را به عنوان ورودی به چرخ‌دنده بعدی داده می‌شود.  
در نهایت خروجی آخرین چرخ‌دنده را به یک منعکس‌کننده می‌دهیم.  
برای مثال اگر رشته حروف "abcde" بود باید حرف a را به e تبدیل کند و حرف b را به d معرفی خواهیم کرد.  
خروجی منعکس‌کننده را هم با ترتیب برعکس به چرخ‌دنده‌ها می‌دهیم.



و این کار را برای هر حرف موجود در plain text تکرار می‌کنیم. اما به ازای هر حرفی که کد می‌شود چرخ‌دنده اول به اندازه یک حرف می‌چرخد و به ازای هر 27 چرخش چرخ‌دنده اول، چرخ دنده دوم به اندازه یک حرف می‌چرخد و هر 27 چرخش چرخ‌دنده دوم هم باعث چرخش چرخ‌دنده دوم می‌شود. حالا اگر شخصی در طرف دیگر از تنظیمات مشابه (چرخ‌دنده‌ها و طرز قرار گرفتنشان و اتصالات plugboard) استفاده کند و حروفی که رمز شده‌اند را وارد کند، به متن اصلی دست پیدا می‌کند.

آلمان‌ها تقویمی داشتند که نشان می‌داد هر روز باید از چه تنظیماتی استفاده کنند. (لازم به ذکر است که این تقویم بخش مهمی از این رمزنگاری بود و باید از آن به شدت محافظت می‌شد برای همین با جوهر بسیار حلالی نوشته می‌شد که در صورت محاصره شدن، بتوانند به سرعت آن را از بین ببرند).

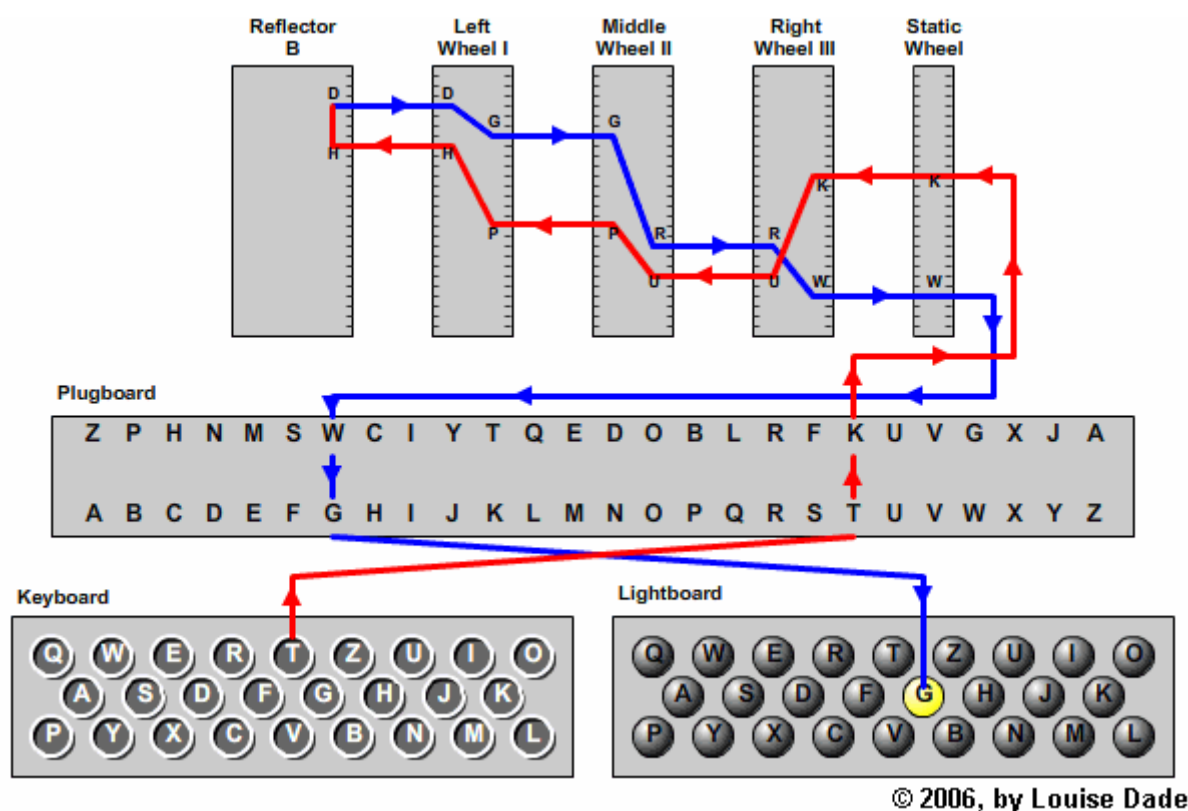
GEHEIM!

SONDER MASCHINENSCHLÜSSEL

APRIL 1940

Tag	Walzenlage			Ringstellung			Steckerverbindungen												Kenngruppen		
30	IV	III	II	07	21	20	AO	BX	CS	DL	GH	IU	JK	MZ	NP	RY	RNN	VJR	SSM	DDG	
29	IV	V	I	20	19	03	AR	FM	GP	HZ	IQ	JU	LV	NS	OT	WX	XTY	YGQ	VZF	TBX	
28	V	III	II	15	26	13	AQ	BM	CP	DN	ER	FY	GI	JV	SZ	WX	XQD	GWJ	HCZ	MWS	
27	I	II	IV	12	16	05	AJ	BZ	DP	EW	HU	KS	LM	NO	RY	TV	GRP	TLI	IAT	GXH	
26	I	V	III	04	15	12	BO	CD	ER	GV	HW	IJ	KP	LQ	MN	XZ	YXU	QSE	FXN	ZKS	
25	II	I	IV	08	13	11	AX	BZ	CU	DI	GN	HK	LQ	MO	RW	TY	TZX	PTI	LQR	MJZ	
24	IV	V	II	14	05	26	BD	CX	EV	FY	HI	JQ	LZ	NT	OR	SW	NVH	HZU	UCU	PUW	
23	III	I	V	11	24	23	AP	BF	EY	HS	IK	JT	MO	QW	UX	VZ	UQA	TRG	SFU	VLV	
22	IV	V	III	13	12	04	AG	CO	DY	EF	IQ	JK	MR	PT	SX	WZ	AHX	FVI	UTO	WCM	
21	IV	V	I	04	06	17	BY	DS	FR	GH	JX	KO	LU	MQ	NW	TV	NAV	YNY	OLD	BNX	
20	I	III	V	24	06	09	AC	BL	DY	FO	GI	HR	KV	MX	NU	SW	VEE	SGG	ZMR	CKN	
19	I	IV	III	10	15	11	AW	BS	CO	DE	GP	HR	IL	JV	MY	UZ	VOW	QZZ	ZIN	CZW	
18	II	V	I	11	15	18	BK	CI	EV	FT	HX	JP	LR	MN	QU	SY	LTK	BPE	VTX	HMD	
17	V	II	IV	16	14	15	BU	CJ	EI	FL	GS	HX	MY	OZ	PR	TV	PYO	BZV	JNB	DUP	
16	II	III	V	17	02	13	AM	BH	DP	ET	GJ	LN	QY	RW	SX	UV	IYP	LVN	BRK	HDJ	
15	V	III	II	23	26	02	AN	BL	CH	DK	EJ	MQ	OT	RW	VX	YZ	HUN	IDO	PGB	XJU	
14	II	V	IV	05	21	10	AL	BO	CZ	EJ	FK	GS	HV	PR	TX	WY	PMM	ZQE	NEU	BXU	
13	III	II	I	17	12	17	AG	BX	CW	DK	JN	LV	MS	OU	PR	QZ	JSY	IRD	TQV	AMN	
12	I	III	II	26	09	11	AL	CQ	DZ	EP	FX	IT	MR	NW	OU	SY	IRN	LRL	SLG	PET	
11	IV	III	V	08	02	18	AW	BV	CP	FX	HZ	IN	KU	LS	MT	RY	PYP	BEG	UJW	ROJ	
10	II	IV	III	15	02	21	AM	BW	CT	EF	HR	IY	JQ	KV	LU	OX	RQD	MBX	DCO	PBY	
09	III	I	V	13	22	08	AR	BC	DM	EF	GW	HZ	NX	OV	PS	QT	GAU	QAT	WZQ	FQE	
08	IV	I	III	04	08	04	AZ	BE	DV	HX	IU	JL	KS	NP	OQ	TW	DFG	OET	YYU	ONG	
07	V	II	III	03	06	06	BI	CG	EF	HN	JX	KS	LP	MQ	RT	WY	VVF	ADY	XYO	DHH	
06	I	IV	III	13	18	05	AS	BY	CD	EW	FN	IT	JZ	KV	MR	PU	FBV	DKF	LIY	QJH	
05	V	I	II	20	20	12	AX	CS	DI	HN	JO	KM	LR	PU	VW	YZ	VNK	OVA	VXZ	ANZ	
04	III	V	II	09	12	13	AL	BV	CI	DO	EQ	FX	GH	KP	SY	UZ	TVF	AWC	QGE	KEI	
03	IV	II	III	15	21	03	AY	BZ	CK	FU	GT	HP	IN	MO	QR	SV	YLR	BFO	CYO	VJV	
02	IV	III	V	06	04	09	AM	BO	CD	EQ	FW	GJ	HU	KR	NZ	VX	BUV	SVE	PRG	DGL	
01	I	II	IV	04	04	16	AS	BY	CJ	DX	EV	FU	GR	HQ	IK	MT	URX	YGP	NBW	VRE	

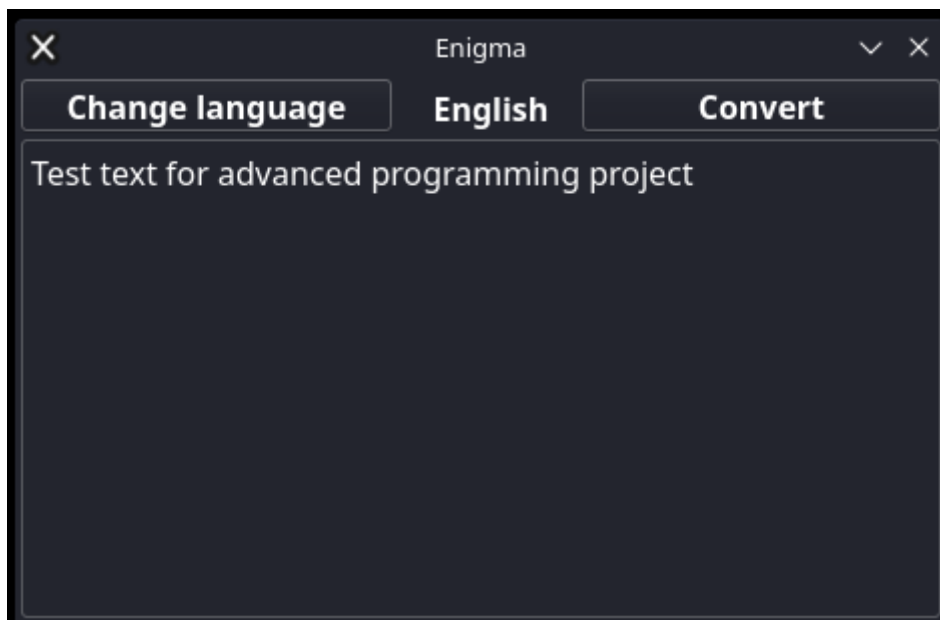
شمای کلی از کارکرد این ماشین:



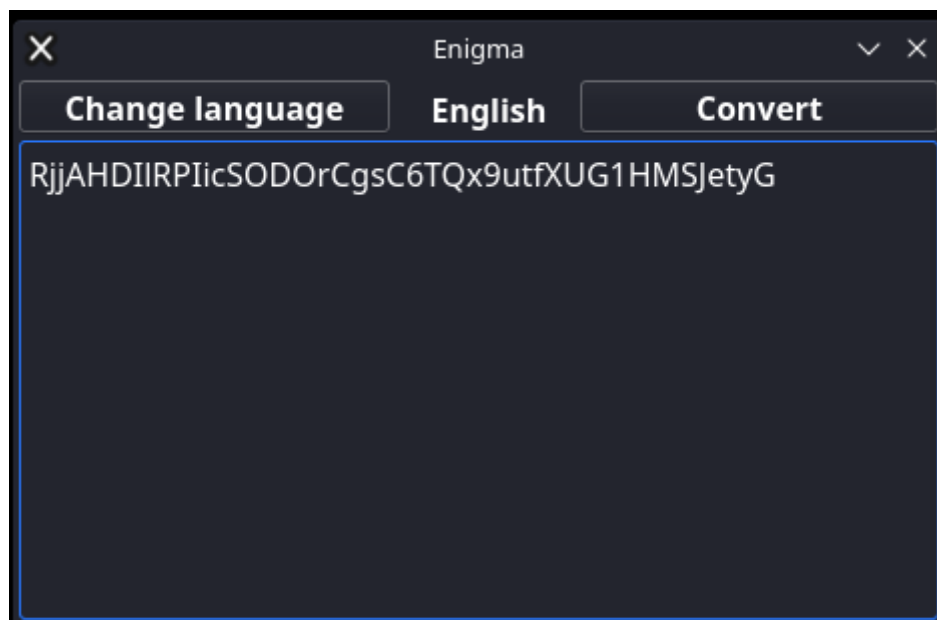
### الزامات پروژه:

- تولید و ذخیره چرخ‌دنده‌های انتزاعی و امکان استفاده از آن‌ها
- طراحی بخشی مخصوص تنظیمات ماشین
  - یا توسط کاربر ورودی گرفته شود.
  - یا با استفاده از خواندن تاریخ و ساعت سیستم، تنظیمات به صورت خودکار به روز رسانی شوند.
- ایجاد یک textbox جهت گرفتن ورودی و نمایش دادن خروجی
- طراحی تابعی برای encryption و decryption
- اتصال این تابع به یک push button
- پیاده سازی plug board الزامی نمی‌باشد.
- پشتیبانی از زبان انگلیسی کافی است. (2x26 حرف (کوچک و بزرگ) و یک space)
- هرگونه شخصی‌سازی، بهبود و ارتقا، افزایش جزئیات کاربردی و دیگر کارهای خلاقانه و مفید در برنامه مجاز است و در صورت لزوم نمره تشویقی برای آن در نظر گرفته می‌شود.

نمایی از پروژه:



متن دلخواه را در textbox وارد می‌کنیم و سپس آن را Convert میکنیم :



همان‌طور که مشاهده می‌کنید متن ما رمزنگاری شده‌است و می‌توانیم آن را کپی و ارسال کنیم. هر کسی که با تنظیمات مشابه، این متن را convert کند به متن اصلی دست پیدا می‌کند)  
در این برنامه تنظیمات ماشین در هر ساعت به صورت خودکار تغییر می‌کند.