**TEHNIČKA** **ŠKOLA U ZAGREBU**

**Zagreb, Palmotićeva 84**

**ZAVRŠNI RAD**

**DIGITALNI SAT SA MREŽNOM POVEZANOŠĆU**

**Mentor: Učenik:**

**Siniša Tevelly, dipl. ing. Niko Pešut, 4e1**

**Zagreb, lipanj 2023**

**Digitalni sat sa mrežnom povezanošću**

Sadržaj

[1. Uvod 1](#_Toc129655423)

[1.1 Uvodni govor 1](#_Toc129655424)

[1.2 Repozitorij s kodom i datotekama 2](#_Toc129655425)

[2. Tehnologije i dijelovi sata 3](#_Toc129655426)

[2.1 CAD dizajn 3](#_Toc129655428)

[2.3 Tiskane pločice 4](#_Toc129655433)

[2.4 ESP 32 mikrokontroler 6](#_Toc129655434)

[2.5 ADS1115 analogno digitalni pretvornik 7](#_Toc129655435)

[2.6 DS3231 sat stvarnog vremena 8](#_Toc129655436)

[2.7 HUB75 64x32 matrica 9](#_Toc129655437)

[3. Funkcije sata 10](#_Toc129655438)

[3.1 Postavljanje konfiguracije 10](#_Toc129655442)

[3.2 Dohvaćanje Geo lokacije 11](#_Toc129655443)

[3.3 Dohvaćanje vremena 11](#_Toc129655444)

[3.4 Dohvaćanje vremenskih stavki 11](#_Toc129655445)

[3.5 Vremenska crta 12](#_Toc129655446)

[3.6 Pohrana podataka 12](#_Toc129655454)

[3.7 Tipke sata 13](#_Toc129655455)

[3.8 Prikazi 13](#_Toc129655471)

[3.8.1 . 13](#_Toc129655472)

[3.8.2 . 14](#_Toc129655485)

[3.8.3 . 14](#_Toc129655486)

[3.8.4 . 15](#_Toc129655487)

[3.8.5 . 15](#_Toc129655488)

[3.8.6 . 16](#_Toc129655489)

[4. Kôd 17](#_Toc129655493)

[4.1 Glavni dio programa 17](#_Toc129655498)

[4.2 Konfiguracija i pohrana podataka 18](#_Toc129655499)

[4.3 HTTP zahtjevi 18](#_Toc129655500)

[4.4 Crtanje po matrici 18](#_Toc129655501)

[4.5 Prilog Kôd-a 18](#_Toc129655502)

# Uvod

# Uvodni govor

Digitalni satovi danas su neizostavni dio našeg svakodnevnog života. Osim što nam pokazuju točno vrijeme, često dolaze s dodatnim funkcijama poput alarmnih funkcija, kalendara i drugih korisnih opcija.

U ovom projektu razvijen je digitalni sat koji koristi hub75 64x32 zaslon i esp32 mikrokontroler te DS3231 RTC sat za prikazivanje vremena i drugih značajki. Jedna od najznačajnijih karakteristika ovog sata je njegova mrežna povezanost, što omogućuje povezivanje s internetom, sinkronizaciju vremena putem mreže te dohvaćanja trenutnih vremenskih uvjeta na specificiranoj lokaciji.

Cilj ovog rada je pokazati da je moguće izgraditi jednostavan, ali funkcionalan digitalni sat koristeći različite tehnologije.

Rad će u prvom poglavlju se baviti tehnologijama i dijelovima sata, a onda će se fokusirati na njegove funkcionalnosti, te na kraju će se objasniti kôd.

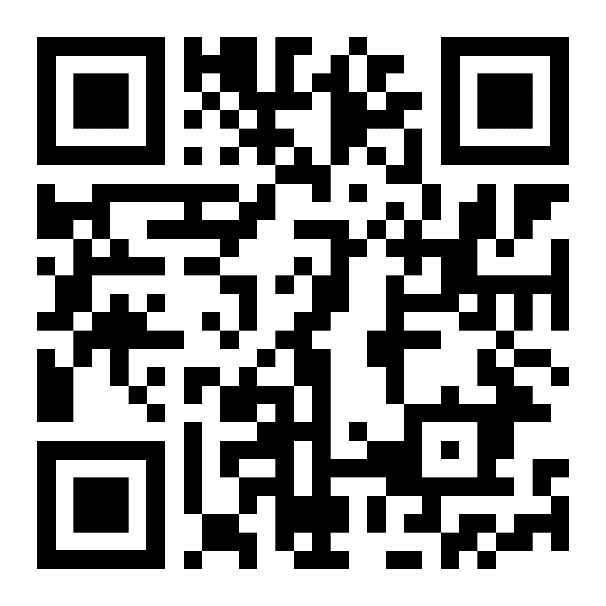
![Slika na kojoj se prikazuje tekst, sat, semafor, auto-radio

Opis je automatski generiran](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/4RDqRXhpZgAATU0AKgAAAAgABAEPAAIAAAAHAAAISgEQAAIAAAAMAAAIUodpAAQAAAABAAAIXuocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAFhpYW9taQAAUE9DTyBGMiBQcm8AAAaQAwACAAAAFAAAELiQBAACAAAAFAAAEMySCAADAAAAAQAAAACSkQACAAAAAzAwAACSkgACAAAAAzAwAADqHAAHAAAIDAAACKwAAAAAHOoAAAAIAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAyMDIzOjAzOjEyIDIwOjMzOjIzADIwMjM6MDM6MTIgMjA6MzM6MjMAAAD/4QmcaHR0cDovL25zLmFkb2JlLmNvbS94YXAvMS4wLwA8P3hwYWNrZXQgYmVnaW49J++7vycgaWQ9J1c1TTBNcENlaGlIenJlU3pOVGN6a2M5ZCc/Pg0KPHg6eG1wbWV0YSB4bWxuczp4PSJhZG9iZTpuczptZXRhLyI+PHJkZjpSREYgeG1sbnM6cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOnhtcD0iaHR0cDovL25zLmFkb2JlLmNvbS94YXAvMS4wLyI+PHhtcDpDcmVhdGVEYXRlPjIwMjMtMDMtMTJUMjA6MzM6MjM8L3htcDpDcmVhdGVEYXRlPjwvcmRmOkRlc2NyaXB0aW9uPjwvcmRmOlJERj48L3g6eG1wbWV0YT4NCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgPD94cGFja2V0IGVuZD0ndyc/Pv/bAEMAAwICAwICAwMDAwQDAwQFCAUFBAQFCgcHBggMCgwMCwoLCw0OEhANDhEOCwsQFhARExQVFRUMDxcYFhQYEhQVFP/bAEMBAwQEBQQFCQUFCRQNCw0UFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFP/AABEIAjoDxAMBIgACEQEDEQH/xAAfAAABBQEBAQEBAQAAAAAAAAAAAQIDBAUGBwgJCgv/xAC1EAACAQMDAgQDBQUEBAAAAX0BAgMABBEFEiExQQYTUWEHInEUMoGRoQgjQrHBFVLR8CQzYnKCCQoWFxgZGiUmJygpKjQ1Njc4OTpDREVGR0hJSlNUVVZXWFlaY2RlZmdoaWpzdHV2d3h5eoOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4eLj5OXm5+jp6vHy8/T19vf4+fr/xAAfAQADAQEBAQEBAQEBAAAAAAAAAQIDBAUGBwgJCgv/xAC1EQACAQIEBAMEBwUEBAABAncAAQIDEQQFITEGEkFRB2FxEyIygQgUQpGhscEJIzNS8BVictEKFiQ04SXxFxgZGiYnKCkqNTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqCg4SFhoeIiYqSk5SVlpeYmZqio6Slpqeoqaqys7S1tre4ubrCw8TFxsfIycrS09TV1tfY2dri4+Tl5ufo6ery8/T19vf4+fr/2gAMAwEAAhEDEQA/APp7Tjut1NWqz9HfdapznirxNfnCPrgNJS5prUxHMfEaH7R4XvUx1jb+VcN8Eph/wi8Skf6t9v5GvSPFUXnaHdLjI2GvKPgnIV0u8hJ+5M38zTix9D35m6GkXnOKP+Wad8qD+lNHytmugxHpxUM2VuF9xipd2KhvMrsfsGoGPDYamtQxO4GhqQhrdKb1p2Dz6U1etMB3OKdmkXp603qOlIBrKMZNIuKVj1pAeBSAczdKTnmm5NO65+lDAiY80icsc0UKTk0gJVbK4FMm+VeaEYjOKJmyvNMCpn8qRuDyacwxUbNWZSFb5kqIfdxTwx6VF0k9KZI4Uo60L96k71QC96GPHFJupGbNACMKcv3cUz3oXhsUAS/w0z6U9vmxTDx0oELx3pd3amUuaBC7sUuc1GxzSr60FEhXaelOXNNzwKVTTRAh60xuelKzc01m9KAGqvUUjdKXnJpGpDImGcGpCvSmbgKd97BoAQ5qJuKkJ6UwnrQA1jSrTG7YpynbmgYwnmnK2VNMb71OU4U0AMHWl9aZu+ans3y4oAbz17VGSakZfkNRD7tAEjN8tRLncaeeOKaq/NxQIa2eahl+7U7cVDMCy0AhFGe1LiheB1oPSgBy/c/GkH3qcv3KYv3qYhJqjXvUstRLy1UAjUxqkbvUZPtSATOTgU+moNzZpzGgBGbnimc05u1J+FMBj9RTG+lPbJxSPQBGy5waU/eWlPal9KABjTOOakbFR96AF9aZtHSndqF6dKAI2UYqCRdwP0q03CmoJPu1LKRFGu2MDFPBPApU+4KWkAj54oWnt933pq8UDEYdqQinN94Uj9aAE/hNMUVJ/DTexoAjbrSMv4U80H72aAK8/wAsMh9FNePapGy2PmSPJF/pBIa8kRmH+6mRkfWvYL47bSY9QEPt2rxq9jaG0iC2FwrvOT/o0ryy9exBwPpWtP4iZbHa/Ey8ub/4e6aktzfahFHbnas4isoV47BCM/iK+efgLCJviZb/ACKojWZsKc4yMde9e6fF61ntPCdm8+m6hBF9mys3iS7kkduP4Uy2Pzrxf9ndBJ8QJJBtwsDkbBgcntXp1NKcjjjuj6lf7tebfHxs/Dm+Gf4lH616OzDbXmXx+YL8Pbserr/OvMw/8WJ1VPhZ8iydeKhJNSyZ/CoWY19ktjwmY+uNuhP0NJ8IRu+IGl/9dKbrx/dkD0NTfBpd/j7Tsc4Ympqfw5ehdP4kfVeqHMfNefeLLOUXdtcNZXMkar/rpLvCD/cUd/xrv9TOUPFefeLo7ZZoZXt9NjcjAnmuC0p9hHzj8q+epbnqz2PevhDpVxe+AtQuZ9C1a5tVX/j51XUDDaJ7hcNu/Svl34pmOPxHc7JLVepEdmnHX1719I/Am30xvDF/NLpWjtOqZW71jUlLf8AhBU187fGB/M8WXH7zepBx5URRPwJr0KekmcnQ968HsZPD+nHk5gQ/N16CuijFcx4BbzPC+nNz/qV+8cnpXUwj8a8Gp8TPSjsX4fuirajC81VhGFFWo+cViUO7U3+dObNNxmgAFNPWn0wntQBC33jUMh+XHWpZPvVXmpoD53/avu9un2EOexP618sP9a+lP2rrjF1Yw7uiZ/OvmyRefWvrsF/CR4uJ/iMQE0UmMdqK77nIfth4dk8zT4iO6g1rVz/g+cTaTbMDuzGpz+Fb1fnO10fVjqQ00mk3cUCKetL5mmzrj+E1418I8pfa3bn5ds7/AM69qvl3Wco/2TXifw1b7P458Q25GP3pNND6Hvtkxaxt93XYOalbAxVXS+dNg+bdxirPpXUYjwaZcL5kZWl+tJncKQDOfLXPWnct16U0MOlLu7UgF45BqI9eKk60xvSqAVaQ0Dg9aac5oARlPXNI3tTqYc1ACfrTqbTs0ARtncRQvU0kjANQM0APWo3+bJPSnKetI33TigCsx+YUhWnNRu2ioAjpki/Kacx5zSH5lIPWgBv607+God21sHpT+3WquAvpSY9Kac5pS3Qd6LgHegg9aM0lMRMM7aZnmiNvlxQVoEIaT+dH3uKbmgQtPFMzRQMkahW29KZzkUudtAhGbJNJu68UN60yQ+lADlahmGKiUkdacTQAw9KcueKRj0pd2etACN6VETipG/WozQMY2Oop6fMrGo2NLG21eaCg60ZwtJnmj+Ek0CYxfvU8/rTFOWp2PagQh+4ai3du9TH7lQHqKAJGpEPWkkPy02P7vFAhGNNYjaB3pzDjOeaY3GKAG5xmgntQe9NzQBKuDHTR96nR/wCrpOA1UgEm9ajQfNUk3pUa9aYCMBUbVIxzTM0AOVdq0xjUp6VEfm4oAGOSKRsU4R80MKAIsdKG+bHrTqRhjFADMHij+IClbtxSAZagBSMA1HwTT3b5aavfNADsfL+FNX7op6gY/CmL93NADG5qCY8VZPaoJlHNSxoI+VGetKPamr90ClFIoVugpop7LwDTR1oAMUjLTqQ0AJjC0ztUn8NMoAaeoprGnGmmgClqjbdNuT8vCH7xwOleIeIvK+y6YsTQPukz5NrIkSnn++y17XrZ26TdkkDEZPIz+leNa5NibSFmOnTRMcn+0/MhgPPorD+dbUfiRE/hZr/FRrWx8PQp5mn2lz9mAKxTrqEvToWAGK82/ZsQv4yv2Zy2234+XHVq9P8AjPfvJouy0msRAtrgQ+HLeRYRx/E7lsj6GvOv2Y4y3iLWH+biFByc9zXo1X7kjkhuj6MavMv2gm2/Du5wP+Wi16Y3WvL/ANoVivw/m9PNUV52H/ixOqp8LPkmRifeoW5NSOaiY9gea+yPCZi65ny2+hq78FV/4r6xPpuqhrefLf8A3a0vgiN3j6z+jVnV/hSLp/Ej6e1QgLXC+JljGxxcadA54O6MvcN/QCu11SQhTXGa/eMsKR/a5IhyfKtrISSN/wADxwK+fp7nqy2Pbf2dXj0/QdRJvPC9izxn99Ksl1fN7LGrnH4rXz38boy/iOZibmYfMd8yiNfrjHH0r3j9nPxDbWcF/Cuqi3kkjI8m10MzXDf9tGi2r+deLfHmzkj8SSzNBMobcRLdzDef+Ag9a74W5zk6Hqfw3x/wh+mkAf6lehz2rr4O3FcV8LX3eCdLIwf3I+70rtYfmxivBq/Gz0o7Ivx44q0DtFVYu1WefwrAsXPFBoHWlzQITt0qKpG6Uz8KYEDd6rSfdqzJ3qvKcYz+NUgPlL9qqZm8QW6/wqgFfP0h3c17h+09dGbxkUz8iKBivEGr7HCaUonhYj+IxNwopMUV1nOfsp8NLr7V4X09/wC9Cv8AKuvrzz4MXHneC9LOd37lQD+FeiV+ezVptH1K2Q2g06mmoGRTcxOPavEPDKm0+LOsRj/loqt+le4PyrV4ixFl8ZmGcGaEEe+CRTRR7po2f7PUHsTj86u1R0Nt1geuQ5/kKvV0mAUnajrQvepArx58yQE554qUDjpSY+YnGCaM4yKoBaa3UUdaRu1MA6mk79aTOKVj7UAJmmt6ilycU09KQC9s0MOKTPy9KSlYBrLmkHUZpWY801TSAcR8xo7E0D8qG9MUCK8h5NR05vvdKYT7VIxp60nWnY4NMpAMkHekR8809unNQkYagCXORRuGBTVoIx7UALkUnalXFK3GDVXFcI+GIpzZXkVH0INPPzDrTENztbNI1Dc4zSUCAU7dTOM0MaAF3cc8Uu/NRls4pfpQBIW461HndQ3T0oXmgA27s4NHsaceATTPU0AI1KByKTIyKVuooAa1RtmpHOBUWeaQxvB5NKv3TTTjNOX7ppgNajGVoNKpHlmgCJfvU5mpnO6j1zQBI0n7sioedwp/BApin5hQAk2doBNEbcHFPm+7TIu9AgfhailbAHPepW5qCb+Ee9AxaTnmjr9KWgRJD/q/ekHWnR/6sdqb/FVIBJckiol+9UknWmAUANY44pvVqGoT71FwHmo171Ix4qNR1pgSHmo2qT0qOTrQAw0P2pO9DH5qAE7ihfvGkx81OXuaAGSU2nSdKZQBIrdajXPI96eopv8AEQKABsVDJ3qX9ahmOMmpZQDhfeigfdFIc8Uhjz2ooHSjNACnGOKY1OpGoAZ1pM/L709fu0ztQAzdzTX45pzfSmPQBmeIZDHol4w352HHljLfh714p4pkiXWtF85ltpG2/vbm0W7k690CnmvZ/FAH9gXY2l8rjar7SefWvI7oXH/CdaLHbS63DIu3H9kQmR15/vEgZ/GunD6zRlU+Bln423H2jTZ1C3l7EtuMzXUUdgucdRGApI9sZrlP2W4R9s16UAceWnB9s4/Wul/aIjdZ7tr17rzhEMP4gnea4Jx/CoLAH8cCsX9lqL/RfEDnBP2hB7/cH+NdtZ/u5M56e6PeGNeWftEceAJP+uy16exxXln7RUm34fv/ANd0/rXn4b+LH1Omp8LPkyRsNUDN0qSXNV5DX2R4Jma1gwSY5OK0/geP+K6tj/sNWPrDH7PJ/u1s/A3nxvEfSNj/ACqK38KRrT+NH0bqknXmua1JZZrRtraptBzttABF9WYnj6Vv6qwVSa5nVLVrixd0tmuAh53Xfkxr7sBya8CmepI9j/Zz1iW8ivbCPW/EbqyEfY9EsIo8+zThlYD8a8g/aAsRp2uYNrDaSszf8fU5mue/J969N/Zt0/UdSuLkJoWvanAiMW2ambaxUerHacivNPj60MWshInsoMSEeVZDzvw8zua7Y357HJ0Z3XwkYt4H03nI8oDgYrvLWvOvg27N4Hsc7uFx83Xqa9GteOeleDW/iSPRh8KsaMIHFTLVaNuanDeprA0Y/NISaKGo6CEyaax4NONMPHSgCB+9QTZ2ippOM1WlYAdadwPjP9oy483xzcDPQ4ryRq9H+OFwLnxzesDn52H615wyk819th1alE8Ct8bG/hRS7iOlFdRifrF+zvefavAOnDP3AUz9DXrwrwb9l678zwVHH/cc17wPu1+e1v4kj6eHwoDSUv40lZFjGHy4rxXxVGtr8WdJl6NJEyfkc/5+te1tXifxQX7P8QPDkwGB5rIT6ZXP9Ka3Gj23QW/0SUZ/iBx+H/1qvmsrw++UlGRyAf8AP51qN9K6E9EZPcKKbu6UbuaQgYHrmkPWndRTSwAqgDFI3OKXdTGY80AJuHekZu/amFsninfeAoAeMUx+uKOBTWbdQAbu1LnrTN3Y00NnvSAc1NHSlY8UzdjPNIRK3aimNJnHtTl7UCKzfePrTfrSyfLIR71Gx+akyh38NRN+tPD9u9QseaQDw3FMdaFalYgrQAxeac33etRBhu+tPY9KABWP40/+EVAGO+pecUEi0K3SmbsZz1ojbtVAPzUdPf5fpUTGgBwpuTimk+9Ju45oESUqtUO80vmUDJtwo3ACot3vSNJmgRLuyKRmFReZ19KTcaAH05u1Rb+RzTsnjNADm9KiyKe/HvUJpMBWxninI3BqPPuKcpwpPQU7ANZsGjdlajkkB7gUK/y0WAUck4o+tIrc0rEdzQMVuF9qjT7w9aWaRFjHzD86hjmRpOHX86ALEhytMXoaR5F5wyk/WmLKvIzz9aBDm461BMfmX61M3IzUUnVT2FAwzS7uKi81R/EB7Z60eYG6HNAFlW+Sm5HFInzL1pqkM2M1SEOfljSfw0jkZ4o3DHNAER70sf3qacY5NNRwuckD8aVgJn+WmL3pjyKf4v1ojYdjVATHoKikODUh+7UDfeoAFobk0qUMaAEY9O1IOhpG60q/dxQA1qTFK3amk07APpvc0opjPhqTAGyKglyV696sNVeb7vHPNZspDkXIFG3mnJ90dqRmxQhi4IAzxSrTGbOKNx60wJMUx6QsetI56UCHdqaKP4etJ24NAhjdaZJTi3PrUb9etSUYnjAf8U7d5WF/lxic4T/gR9K8/XT31rxjpEUdg2qzEIPs1lIbdTjt5nltx74ruvHMwXw7OG+zYJA/0s4j6/xVw9jGuq+ONNtvJ/tlVC5tt4gg98t6V10L3ujOdramT+0ZINLW7tBHp+lTNhGsbRBcSdOhk2jn3wKr/s125t9C1wn5Sb9lK45XEacE1H+0FeRKLmCK6tbMLIF+w6bGJkOO3mAdc1d/Z9xF4MumHG68l4x0xgde9dVa/sjCFuY9Qkm968p/aMm/4oE+v2hP616RJcfLx0ryf9oS5DeCgpP/AC8Lj8jXJhYv2sTaq/cZ8xufWoJWPFTspKgjJH0qKSNv7p619ijwTD1ck28n+7W98C/+R0Q+kTfzFYesRP8AZpflbpnpXQ/AyJk8X7mUqohblhjuKyr60pI1p/Gj3nWGxmua1HyWs3MsVi23JD3ZO5eP4R3NbmtXEag/Oo/GsCXUEjs5w12sBIJB+yCZycdAcHb9a8SEX2PSlJHU/s32VjceIpZrrSNEvSoYrPrWqrFGv0iBBNZf7QjM18whmh8rzduzTYysCjPRS2Sfrk1rfsuatZxeLmZ9V8P2MzFgjXGlT3l1/wAAUKyg/UVS/aKk+13d4Xe4vG8//XXEYtQfpGOg+oFdsbKpY5VqmbnwRb/ihLP1+b+LP8Rr0y27HtXlPwHkDeAoMbfllkHy9PvmvU7djtFeBiF+8kenT+FGhH1qf8KrRt7VYVq5TUfT+tRhqcuc0Eg1MYdTUhxxmom+6aAK8i571VmAGfpVqTiqF/II7WZz2Qn9KuKuwZ8JfFS4W48YXzLz+8P8640/St/xxJ53ia/YdPMNc6z4r7qnpBI+dqfExvSilb6UVpYg/Sj9km/Euh3cGc7XBOTX0wvzKMV8kfsiXRE2oQnPY19bR8xjjFfA4j+LI+np6wQuaSlb0pua5yxO9eK/GyMw614fuAcbLtQfxBr2qvIPj1CFsrCZvl8u5jOfTmqg/eQz1PwzJujz/wBMh1/Ctlq5Lw7qUFjpa3dxIEhjtyzMTwABk/yrz3Xv2iraaIJo8SvvG5ZWdWO31wDx+NddChOsvcVzlq1YUnebPamYKpJIA96w9U8baRpbFZLtZJB/yzh+Y/Tjp+NfPWrfFy91SE/aIpblCc7GnAXH0A/pWbJ8QJ/4NHX2/wBNBBH/AHxXcsvrdUcTx1Hue7XHxbg5+z2EhP8A01cL/LNUJPi5c7SDp8f/AH8P+FeLTfEG5DDGlRj/AGvtGR/6DVaT4hXasc6dEP8Atr/9atfqNTsR9ep9z24/GC5OMacgx/tmj/hb15tONLUn13HFeJf8LCmaMH7AvmY5XzeM/lUK/ETVBKM6bGIs/wDPTn8Oar6hU7E/XqXc9vHxavtw/wCJbF7jcaX/AIW9dg4GmRn6yGvF/wDhO7uaBnNmvmDp8+FI9PrVT/hOtWZQ0elxMx5YGbH5U/qNT+Uf16n3Pcm+MF2G2/2ZH+Mp/wAKD8Xror/yDYN3/XY//E14a/jTVyx26ZEMnvJnj8KF8cXm1t+mgycBV3EAnPPIHTFL6jPsH12n3PbG+MFyjc6bB7/6Q3/xFH/C6J1Y50eEgf8ATy3/AMbrxKTxdcrAd1lGzljgiQ42+/y9aqv48lTKtpqrgHayyk854yNtV9Rl2F9ep9z3U/Gq55B0WEj1+1nP/oumr8aZlX59Ij/C6J/9krwSP4hXW0Z0tTwc/vCP6Uknj18oVsMg9QZMGn/Z8+wLHU+575/wuyReukJgd/tH/wBjTv8AheDkDGj9f+m//wBavCP+E4Hlx5tsFuu0g7R6YJ5P403/AITiFmIMEq4JxwBkfn1o+oS7B9ep9z3Ob40k7j/ZLe484f4VH/wug550ojH/AE1FeHL42jWRt8UoiJO0hMnHvzSw+MrVl+dZASe0ZPFL+z5dh/Xodz3FvjOP4dKkP/bVajPxoGRu0uQD/rqteOR+KNNCOXuZAcHaPIf8M8UybxZpaqgN1l2HIEEmB/47S/s+XYPr1PueySfGiMZP9mS4/wCui0xPjenAOjz4P/TVK8aXxdpwVwZt5ABQGNxk55HK+lI3i/SFkOblwmP+ebbgfyqv7Pf8pP16Hc9p/wCFz2+4n+yrjPr5i4pzfGqILn+yZSPXzhn9Frxb/hLtEZlC3UzrxkiI/jT18XaKqhRdSE98xAH24zS/s99g+vw7nsi/GeBhkaZL+Mo/wqRfjVEB82lye+JR/hXi7eMNKU/66dAT18kHH/j1C+LNGkkAa7uUj3AFzAAMY5z83HNH9nPsH16H8x7O3xrtTjGmS/jIB/Smp8ardWz/AGXKfpKD/SvEn8a6G4jZZbpzuw4WFeF9R83Wlg8WaVK7L5k6A/dLQ4Hvnml/Z77C+vQ7nuDfGq2dMnTJ/wDv4tMb40Wu3P8AZk//AH2teKxeMNHVWEtzcIRk4+znH55quPGWjHaEupCer7oT79MfhT/s99hfXodz3BfjXaSYH9l3I/4Ev+NOX4yWjHH9mXGeuNy/414b/wAJbpaknz5Ofu/uHP54FWW8WaPHtJvXCEfNm1lznvj5af8AZ77B9eh3PaP+FzWS/e066B+qf40jfGayXrp91/30n+NeM2/ijw9Kx3avMARyv9nyEg/XPNTf8JF4bVfl1y4IPBDae4/L5uaX9nvsP69DuexH4xWe1T/Z1yAwyDuTpnHY0w/Gaybk2FwP+BLXilx4u0ZlJhvLi5IHGy0KZ5OcjJPTHT3pH8SadtBjuJnbHKSRDPfpz9O1P+z32H9dh3Pbl+Mlgow1jOD/ALwpw+MlhyfsM5/4EK8Hbxno5XAN0ZAeQVCAenVagHi62jZkkEgYjKkDAHPfNH9nvsL69Due+/8AC5dPPIsrj6ZFO/4XVpeBm0ud34V4JJ4y02OWNBK4kkXKKyE859RxjFWf+Ei09Iy/2j5twwoAJ24579c0vqHkP67Hue4f8Ls0s5/0O5x7gVx/xP8Ajotr4P1BdHiu4dVkQLbsqA4O4Z/TNeew+J9OkjR2d15+b5Dz9KqL4ssTEqvI5lz82yByuPY4rSGBs72JeOjbc8huPjR8UWuiG1rUo+fuLGP6qa1tK/aA+IVgzCa4l1DchXbcrJgcdfkZea9DbXLCSYBJY1U95EcMB642/WoX1ewaORhLGZFPAdH27fUkIf5V6fso/wDPtHI8VH+c8lvPi18S5pWlTXNYgRjwtuXCj2HGat+D/i98SLXxPpkk+sa9eWy3CNLFI0jKyhhkHIxXo3/CVWG3aGhkTOcRF2H/AKLpq+LbC3UM6KvXDK2APTqtU6ScbKCEsXFP4j6IX426c2Smm3bAk7RuXP5VyXxK+NzyeDNRi03Tr6C/mTZDIvzEcjPQccZryn/hOoNyRjZK6rhnhdCufqD/ADqVvGkMnyNHMBgcIqtk45Od1eZHL3Fp2Op46LW547cfET4ktMC+oatFk8LGGA/St3w98WPHulea08l5fllwPtAY7Se49676bxVaNlYt28c/vIgAPyaq/wDwlC+QzvCrMp5AGM16Spr+RHK8VH+Y8uu/iZ8SPOfOsatGAfur2/Srvg/4kfEW18S6bdT6hq1zbRzo0sbnhkDDIPHpXoC+KQpUtbbjnLAYyAe3PemL4wZ5GdbIxrGMMzFTz+dEqXMrciBYqN/iPf4/jpFMv7vSJtvYtIorlvil8YtR1LwTfWukafc219MFEc0MnzLhgT0HGQCPxryp/Hk424s2ZeoXcM8ewNRyfEW7kwn9nsqcZ5XOPrurzll/LK9jf6/C255c/in4lTTsz6hrpPXCvIAB9BWvpPxN+JekwzBLi+mEi7WN1bGUgf7JYcH3Fd3d+LZGbMARsEKGzwPlGcjjvmq0fiy7kkKRx2u5R85B3/8AoP8ALNep7O/2EcrxUe553N40+I0uW/tHWVBPGHkQfzra8B+NviNpHjDR727vNXmsY7pDNHNKxV0BGQcnpiurj8VXpBXyYA6sSPlYY9M804eK79VnzFAm4/u1VSQD7knmh0bq3KgWKje9z39vjoGxs0pmY/w+Z/8AWrjvid8atavPC1za6TpVxa30mNs0bcrggmvMpPFmoxtgNZtkcoI3GKgm8aauzLvW3YFgDnIOBx0zXBHAJSvY3eYK1rnDyeMvibNIzS6jqikH7qk4Fa2l/Ej4lWFrNEk1xciRdpe4jLOv0OeK3v8AhI9RUnbLbkHB+XJ+tOHiK8kjdfNS3ZmyJGAwB3GK7/Yr+RHP9bXc4WTxh8S7h2X+0tUQjPRyorc+HfjT4iaX4w0u9u7/AFOezhmBngu528uRe4INbNx4g1D5fKljcqMH9z1PqOabJr2pqo3XVuA/zHbCSQc9CN1U6EZK3KhLGJPc97Hx61BgQuj2/wAvGfPY5/8AHaiPx41M/wDMItkJ7tKxH8hXg58Sa0G3faYfLB4C25yR1/hqFPFGtyTIhubZnbhUkjww98HnFcX9nLsdH9oLue+H48auDkadZKPq7f1FVpfjlr27IttPA9DFJ/8AF14PN4o1lZnjF2Fb7uz7KwJ9xlcU5tc1iRQXdY+3zRg/yFL+zl2D+0E+p7qPjdr75Jt9PX/tm/8A8VUsXxs1jH72K1UH+5GQf1Jrwb+1tXuGdmvUAzxthX+tJLqWsYw1+ucZXdGuOnsaf9nLsL+0Ee7v8ZtXKgg2/wB4gfu6ib4wa9g7Wt89sxivB/7Q1lo2b7Zwp5KQjFL/AGtrEu1FvcMehMYqllyF/aB7s/xe8Qxsm423P/TPt+dEnxa1pnUo8PJ/55//AF68FbUNRmxI2qzIVB2qsQwTTI9R1QbR/a0xGefkT/Cn/Zq7In+0D3w/FjXip/fwg4zgRf8A16Y3xQ1/yzvuYsZADeSMfzrwm61C9aTzE1S5ViMbVVAB/wCO9aiW6vmykms3YjJzn5Pl/wDHc0f2dHsP+0T32P4meIZWAXU7dQoLH9wucfj1qMfEfxSUZhqCAKTub7PHj+VeASTXTN8urXwGe8uD+IFNf7W6lf7SvGOepuWB/Sj+z49kL+0D6Ab4oa8YwX1JUyOGEEf+FVW+J+v7Sx1UOOgxEi/yFeD7bv7x1C8bA4zdN/WmgXDYZtUvjIOm6XgU/wCz12Qf2ge6/wDCzddk2x/2s455O0DFVZfiJr/BOsTEdgMV4vJ5sg3Nd3Dk9dzmiS3ZVwl7PgjG0uTj8af1GPkL6+2e1x+OPEE2B/bM6Fu24AfyqqvjbXVmkDa3dy8YVQ64z69OleN/ZnXBa7lYKM7RKeaGUCNlWadQ3UmU7vpmn9QiH15nsH/CZa8rYbWrsnofmXj9KSbxbqy4J1m4xnlvN4rx5odyswmmJ44Mzc0jRRyLtKlO5KyN/jVfUIE/Xmer6h4l1C+h2T6lLdW5AJR3yB6fjVCbUJrphLPdszj7vmOS34V5imn2asTNE0gPX943+NQ/2bpqy7pLZZVHQFzVLBRWxH11s9FvtUkuIxDNeiWJm4iMwIz9KdZ60+nwiGK9FtAuT5KT7R7ng153/Z+m/OUs4/Ybjx9BUS29rHuC28IU9mjBP54zV/VYi+tvsekTeKmbLm/coen78nH61k6lq8V9EqNeLMByVklDAN+NcS1pbMpARVB6gKKf5NvjaAqr0+4KpYWKdxPFyZvzXkU8Z2zwKAeSdv5Cs+d7LyQ4voSScFTjdWebO1+8wXI6fKDQ0dqpH7uMEdxGMmtlSRl7d9gkaG6Y5ljZcYGcAD0qD5LfBjuU245KHGPbirCtbxknEbcYwVFJ9qhWMxiCHDdT5YzVeyQvbMrboWIAkTnn72fzqaG4t4MpvhXnnLDmka4iVQoijA9AgFHnR4yFQH2FL2KK9vI3PDPjK68G3y32ha5JpF8owJrG5Mb4PbINV9f8UT+JpZLjV9Rm1aeVgXmvJWlYn/eashWijbdt/wC+f/1VL5kfBdM/VqPYR6h9YktjoNJ+IFx4XtVttJu0gswc+XJGGAYnJ5rft/jtrFvwRZXIHU+Uy/1rgGmgjYAquCP71It3bjICjP1rnlgqM9WtTWOMqx2Z6zYftHBGP2vTUKj+KFyP5itqH9pLw8NpmtbmLPXHzV4Z9qhzlk3cetN+0xR4eNEY4yAwyK53lmHl0Nlj6x9J6F8cPCuvXiW0N3JFI3QyxlR+Jr0BWz7jFfFWlxz+d5yvHHI77Sx+VVXux+gzX1X8MdeufE3gvTdSuyrSzqxDL/EoYqD+IFeJj8FHDpSg9D1MJipV21I63qKhk6Gn5xmo5G+X2rxT0yBs896yPEUgh0W/cnAWF/8A0E1rM3Fc348m8nwrqbE8eSw/StKes0hPa58IeKH8zWr18k5kI6VhN+tbOuHdqFyQcjef51kswC+9fdw+FHzstyPzCvGaKUANzRWhJ96fsm3hTxBcxZxuTp+NfZVu26EZr4e/ZcuxD4y2H7zJX27anMIr4fHR5ax9JR+AnNN4oJpK882A15V8fov+KTlkxnYysPwIr1Rq88+NVt9o8G3gxk7M047oZw3xX8SPo/wJuriKb7O80CQeYO28gGvk3QPF0/hK4giv2F5oN0+2O4UZa3f0J64r6P8Ai8jah+zqqqNx8yDI9g2TXyOscFus0MsIktZhsljbPI7MPRh1FfWZOkqUr9zwsw96aTPdIb7bcCNlJRhuSRRlWB7g0SXTB2CjGK8t8CeLpdDuovD+rXKy2Uw3affFuMZ4Uk/y7GvUt0sfyTr5TAYDMNoYfWvesfPSjysja+l6HeR3/dsf5Covt7bsFXB9Njf4UrTbnMcUtu59PP5/LBqvLeR27FXubVWHrc8/qKWhJaa4+UHp7HjFN83cvHPPbmqq6pDxuurUfSTdS/2lApJF1bNx/CG/OgRYF1OikhXC9+OKja6kDZ+YZHpUT3kTcfaY8+wbFM+0xvz50bkcDAI/pQhEh1Joshhilj1JmIUDfnpg5NQ/aE6Fl3emRinG4Q8NOgA6A5/wo0KuyyLq4ZiDG3XoEf8AwqKWaZskpIfX92w/pTYpYdpYyx/VT/8AWprXlqx+a4RR3ZgSB+QzQJkcl4sZyRg9MHj+dRtNu52sPTjipFuIwzrHdQsB82VDj8eR/SmO8cqk+fu/3R1/NaYhi3D8kK/oeDSNdN0+Y/8AAaRljiI3IkuOdrqRn8QKiVueQAP9kNxQIkkujtxg4z6UkVw2SMsB16U3Bx0P02mpI/MVgQCpAyM5FADfthRTtPPfIoF1833hnGelS/Z5mwFVWzz/AKymbnjYgssZ6cuKAGrf+XnDRjjqy5qNr4sfvqc+h/xp7SI2cuv0EyAn86icxRtnzFAPGfPiosIVbpl5H/jtPj1IqyhhuGeVcjB/DNRfuOCs0WP9qZD+gH9aWNsZ8uWNjn+GVf8A4qgRL5mDxyM8cULMrFw3Qj0701nynzvAp/67AE0hlVsDzI34xkSDigRUaOzWQkxKGByD5Z/wq1DqEa/xlWA6lTx9eKQ7mXghh/vcUqwuxGw7iRng0CuRzX8UzHc6uO+5ev6Ukcltn5I4/f5R/hT8vHw5wfTvUm5mB3lFQfMWJpiEZ1MgxEGXPoBUjMFjZjGqBT91cEj8qZGsMnCeW5Ay27H+NN3KgKrJECOSPNVaQiT7QZDlt5YDGHQ4A7dqi3KrBiu1u/FSLMJMBADnusyYP60kjKgG5Ao/2p48fqaAIGeNsk9R3O3/ABpjXP7v7u/tkKWI/EA1Z/dtu3Bj8vy+XLFjPv8ANUL7cbQVQ/3XZc/1p2Q9Qiul8sfumLZ/ijbH/oNOW43cCM8cY2n+opYZPJ4+1WcPOT5hQ/0zipI5I5Mk3cEjdflOB+lFh3E81A4bATPHUCmbYGJYRLkdGVAf1qb5GK8wyep3gj+VI2yNsLt/PApWDUjfaeX6dAWFRGZFUYYbPyqfbuf5WBP+yc01vM4BMjD0BI/mKBalW4kt5VKyrBKjcESKOfzFRQywWqkQpEidCkbqo/LIqy0zIpBaWHnruQj9CT+lNS8Usdsyux4Pmbgfz4FAD/tm+MHymZF4/hx+e6mSMjZHkttY9GVDn9aNzx/KIo2/3bgYP5NUrJ5cigSQS8At5c7/AC+q524z9Mj3pWHdlNrCxt23JaJbFv8ApkBn8RT1+zxqAm1BnspFXG34AWEIqnO9XZmP1OMfpUPzs2VK5zk+ZnP/AKDRbQRAZE2nYqtz2FQtLubiIZHPX/61W5oWQ7i0aKeQu7n+VQOn3906jfx2qkLUia43gLsZs89c0yGYJ8oj2L1zxxUkbrbtuS5j3/3t5X+XajciqD5kYUDB2yk1QhGmj84SrGivjHmMBk/j1p/2kyOMlWK9Dv6fQ0rXUPBFwmD23ZNRtcJ822SNvcDNAD7i+klk3PJuY9W3ZOO3PeoZLtmYEk59zk1HJcBlHKMf91v6LTJcbl/dQgY4GTz+UZNAEjSN1HBPpkVI07KpBKgdec5qHYdpDKsYx/yx3n+QFRlouctcDJHIeTb+R/xoGTtOeM8+vrSCZXY/KC3uCPxqJZoUHyySEf7WSP1FWGmSQACSNSOgYg0rANQ4Y4Xk+hpN+3JYYC805I5ZgzLNbsQe4PH61C0YjzuMDe6l6YxDOWwyOGbOfXFNaW5bI8tSpOSdrj/2WhvIjbO7yxjlhkj9aRWtdpH2/a3o6r/8TVWJEO+Nt21Qp6Ybn9cU953KkGORlI4wyH+tJHNbqP8Aj5Vs8dP0zmkE6LlgckcdyaoBYZH3HG5DjjgMD7YzxTXkf7qBWIzzyPw7igzICfvOxH91uDUaqrLkyyRnOAGjP88U+VjHIJZo94EZGcE7zgH/AL5pvz5BGGxydppyeUmT5rM3/XPrSN5bKxVywU8jZj+tKwxGlfaMNgd+KZKzcYznGW3etOWSLy8kSAZzuAP5VFJcRHBZ5H/3qqwhWd2QbQuPXFNLTq3yAE9e2P1o+2R42lWAPJ5oe6h6BGY57df50xkTebGMsQSPx/lUbXGVPIB696mFxCWMbLKueTlDkflRJdwLxl8+uf8AE0gIYSJGQCRck/xdP51Kscm1mZox6Zds/h8tNW+RsgMxQdjKF/TNNa9jznyn6YyJEH9eaVmOzHHzQxUspHsxP9KGyGPzbsdO1Qf2lb8jDYH+2v8AjSSahZlhhWUY5zL1osFmSyTE9segHakUsynqW64qu1zZSE5GP+Bj/CjztNiZSR5vr84FBSTLMEhYgNuUfSpFjMm7AIx2IpF1TTPvIqpzjyzLTZNa07qWU8Y2+fjH6VJVmKYz2NRSKWj3Bm4OMAHNQSa1p/mALbocDkm4bn9Kjk13T2U+VEq5P3TKxx+OKoVmWJPNVuXKnHTaRULjbgbsmq8mt23l8xwsOmWlfiof7cs2UjZBn33mmvMfKy55bLyGB+lKsJbPIH41T/4SGzHBgt2XGOEc/wBaYfFFrGMJHDt7r5Rx/OkHKzRktpI0R5EZUYZRmBAb3GRyKiKjjkfnVJvFEcyruEeFG1B5RIUZzgc04+KrZUIMUZGeqxYNIfLInn8sfdY9PWody8YbJ9c1A3ii2aKUG23SE/J8gwPXNV4/EFukgZoBgfwqn/16dmPlZp28UbnJGOM5wTStErZbcP8AZXPJqi3i+Paf9GjbPrGOP1qI+LNv3YY1PbEQ/wAaB8rNJotoy20D6801pIkGA0Y/KstvFsm/fsw/97y1/kaRvGl0xzz04xFGP/ZaA5GahZHztlU+yjNNMKr/ABn3OKzP+ExvWIBeRgOgCRD9dtMPiy43ElZj9HVf/ZaQ+VmplCcA7iOfl5qSOTdnapJ+lYzeLLkkNmYH2cf4UN4vuuzSZ7/PTDlZquSx7ZU9KSNXmkEUab5W4CgdB61hSeJplXdtIJ77qVdbuNUZbGB5IZHIa4l3YEUfcsf6VJSiy74i1ZLPw7MELCS8Jt4Fz/yzH33/ABPH519Z/BOL7P8ACvw6h/59s/mxNfEXibWl1vVGkiVY7aBBBbxqMAIoxn6nr+NfdPwtj+z/AA78Px46Wifyr57N/wCHFHuZfGzZ1meKbIcLSbsUx24r5U9shkauN+KV0LXwXqb/APTOuwkPTFed/Gycx+A78ZxuwtdGHV6kSZfCz4ovpd1xKcYBY4rPPPNWrxf3j565qmTzX2y2PnWLiimZorQR9jfs73n2fx9YjJXedvAr72sG3W64r88PgtfCz8d6YxyA0gXiv0I0lt1qnoRXxeYpRqpn0eH+Fl1qTNK3T3pq15RuB5xXHfFSHzvCN+Mf8sm/lXY1zvjy3+0eG71PWJh+lUhnzz8QLkzfs2zkjJVoxwf9qvlGRwynI+bHrX1L4ijN5+znrUZYhoSp57YcCvlORmVnGOhxmvr8o/hzXmeFjvjRas003VLV9N1Zmt4H5iukXJhk7E+3rXT+H/HGqaHt0DUrhbCWFB9lujEskTqehyeQD61wUjfnWnZ61BcaadK1eNpbHP7qeMZltj6r6r6rX0SjdHkNc256tb+KdXaQ2mrMzSAZQpKIty+qsgBI/Gp49Su+ifvO+Z53c/mTXlGneMLmz02TT3EV+kTYt7mQMskYHQryPyNadr8Sr+3Xb9jtZeOrBh+PBrBp3J9i+h6IdWvyDmOP1znpTH1bUt5KpGOO2K4L/hZ18c406yA/4H/jSH4lX7Z/0G0H03f40yvYyO9h1XUSTtCA9ccYqQa5qCxtwqt2wSK89X4jahtI+y2vP+yaG+Imot0trX/vkmmL2DO//t7VVzslyOn+sIpP7c1jg/aX5PGbhjiuAX4hangjyLTB5x5Z/wAaB4/1E/8ALtZ57/I3P/j1CiHsJHfR+IdYViVu51wev2pwP0qP/hJta3f8fc+c8j7U5zXCP451F8f6PZjnP3H/APiqjbxrqROBbWI758t8/wDodUovsL2Ej0B/EutdRdXAyen2l8D6VF/wkGr4Obi4Yfxf6S3P6Vwx8cajtwbezI7fu2/+Lpv/AAml+efJtR9Eb/4qnyMXsJHfN4g1GSEnzmG3HDTkk59PlqI65qfVpZDkdd//ANauIbxpqHeG2H0Q/wCNH/Cbajgjy7f/AL4NHIx+wkdo2qXbNlpGYkcbX/8ArVC19cuvzM7N2+fmuQ/4Ta/4/c2//fFJ/wAJnfhs+Tb59dtHIw9hI6kzXfUo2P8ArqR/WjzLuMbirMjDbkzNxXLHxpf7s+Tb/kaa3jS+P/LG3/75P+NPlYvYSOlZrjaSoceuLhx+tJ516uMxPkjq16//AMTXNf8ACaXoX/j3tv8Avlv8aYfG1/8A8+9r/wB8v/8AFUcjF9XkdSst2efLkXPpfPjP/fNI81y0e0wsPf7Y2f8A0H+tc0PHWoAYFtZ/98P/APF0o8dahwDa2R/7Zt/8VS5WH1eR0BkucBdrN24u2z/6DSH7QGx9nRv9o3Df4VgN46v/AJl+zWe1uvyN/jUf/CbXpz/olpz/ALB/xo5JC+rzOjl+0qxV4FYd9s5NC+bwRbquBj/WtXNL44vlbIt7X/vg/wCNMbxxqJ/5ZW2Ow2U1Fk/V5HUxvLuJMGffzmqaMSMrZjOOpxcMP6Vx48calkfu7b/v3Tx441JRjy7Uj0aM/wCNPkYfV5HW/wAWPKbaOcm4Of8A0GpPM2oAsHJ6sblv/ia48eONS2jMVo3P/PI/40HxvqZ6Q2Y+kbf/ABVHJIf1eR1yySM2TFu+tw2P0FBkdlOYMt/s3D/pxXHr411POTDZH/ti3/xVA8ZamM4js+ev7gn/ANmpezY/q8jr3aVVObdu3P2hmx+NOF7KuMW44PH79/8AGuPXxpqSncI7UN6iE/40f8Jxqvf7P1zkIf8AGnySD6tI6+S6kmLb4Q3rtmemhlVSBEgGOu45rj5PG2qyMSTASfWIGmf8Jnqn96DH/XIUuSQfV5HY7tjDhcYzyaRpn6+RE3/AiK45vGWpbgT9nLDv5QpjeLtSbtb5/wCuf/16fKw+ryOxZ5tw22iEe0zj+VCXU3ObZsr1C3ko/pXHf8JdqWAMwfKc/wCr/wDr03/hLtU3Fg0OT/0zJ/rS5WL6vI7ZdQdzxBIrd83sn9RUc15LJwq5/wB67fH6rXGf8Jfqgz80A/7Z/wD16RvF2pt1W1P/AG7qaORh9Xkdgs11uOFgbnjMjn/2WoLhbpvvW1kwJ7oc/wDoVcm/izUGLHbaq3tax/4Uw+MNTA4a3z6/ZY/8KXKw+ryOn+xyRqNtrZoDznynx+j0mJkwUjswccELJ/8AFVy3/CZat/z1hH/bun+FKfHGsMoUzxEAYGYE4H5UrMPYM6nzbscbrcc9g35dacJLn72Isn0WuQbxrq7MWM8Rb18laibxhq5OftQX/djAo1D6vI7RXkwoZIz7bOf503zpVZxtiGRjlD/jXFHxdrBzi9P/AHwv+FIfF2sbf+P4j/tmv+FKwfV2dssk+Dh1HPA8s4/9CoMl0xG6SNh6eV0/WuF/4SrWMgi/bIOc+Wn/AMTSDxTq+4kahIGOeQiDr16LT1H7Bncm4uVXaDH05zH/AImmtcTnG1YAMY4h/wDr1wv/AAkmq7dpv5No6ABR/Sl/4SLVcY/tGbH1H+FAvq7O6FxdxKx3RIuf+eQH9aj+1TtGWMyle5EYriG8R6txnUbg9/vmk/4STVuf+Jhcf99mgr6uzuGuJW3FbsbQO8YH6U0TXbcLNGwyM4RSa4mHxHqCzK7XUkgBBKsxwR6VHN4g1CSZpBdSJuOQqt09qYvq7O6lmuI8fvBuz12A1ErTsvMp3Y4xHxXDrrN+vzC7kH40LrV+Ol5N7YamP2D7ncGS7Gf3zof9ymH7bwftUqqTjgD/AArjP7c1LH/IQuR/21NL/bWpuOdRuz/23b/GqSD2LO0EV9tVRdTkEZAXb/8AE0klpfbR+9usYycKMfyrjF1XUWGPt92PpO/+NNN5eng3t0f+2zf41dg9h5nbJY38rfILplzwORj9KRtP1FVQt9sQc/MScfpXDNJO/DTzMPQyMf61FsZics35mqs+gew7nctY3rcH7WTjuzU17G6XOfP3ZxjJriPLOOSx/E0nlsvRmA/3jTsw9h5nZfZJ+7P9N1MezlWPPmlT6F64/a/Zm/76NNZGPVm/M0ah7HzOu+zyI3zM7DPZ80x7fdnasvrnca5PYexJ/GmtDmgfsfM6drJjuZi7HuWY5o/s9WxuwVPZpcf1rlmi9R+lJ5I9BS1H7HzOs/smMRqzPbBSSMG6j3fluzUTaXbpJjfaEkZz9pj/APiq5by8npTWSldh7HzOqW0t0f79ngH/AJ7R/wCNNZLXvPagY5/eD+hrlmUelM2j0qW2P2SOpka0jf8A4+rfI9Gz/SmteWi5H2uE89Of8K5jbjtS7RRdj9ku50Bu7PAH2iLp60xrqz4Jmj/CsML14pdtO7H7NG015aAcSqT9KRby1ZgpuI092ViP0FYuw0nl+gpai9mja+2WjMc3Ea477W/wpn26zyMSj/vk/wCFY+zGOKDH7c1Oo/Zo1/t1ru/1xA/3Caa+oWZP+sbGe0f/ANesnZkUnl+1Go/ZxNT+0LRepf8ABP8A69I+pWu0gbyfXb/9esspRso1D2cS+dRg7bh7YpjahB2Dn8BVLy6DGfSlqHs0W/7Qh5+Vx+VM/tCPGNrflVRozjimbD+NLUfJEuHUU7I36Un9ox4HyNn6iqTKfrSbaTbK5Il3+0U/55t+YpDfLz8hx9f/AK1U1U0EHgVN2PkRb+3Lg/IfzpjX3oOarMvtSYNLmZPKizDfKJkjKqJJDtEz5Kxf7WB1q3dXEVtbSWljLI9o3zSSOMNO/wDePt6CsyEbGLY5xxTpJP3Z9qpEtW2KUfL/AI1+hngiLyPCGjR427bSMf8Ajor89bMGS8hX+9Io/MgV+jGhQ/Z9FsI+ywIv5KK+azh/Aj18Ct2XWO2mSNxinsd1RSd6+cPVIGbk15f+0FdC28CyDPLvjn6V6c3fmvF/2lrpY/CcMZOSXJrtwqvViiKr9xnyXdtuYkdap5qW4YE9eagr7GJ86wopOtFWB9MfDa4+z+LNObpiVTz9a/Rnw9IJLCJuxQfyr80/Cknl6xZvnAEq89+or9HvBcwk0W0bO4GJcH8K+OzL44n0WH2Z0DUwd6c3rTVrxjrHelZXiKMTaTcL6qRWpVLWF3WEwHXaadxdT5pmtmvvg540sehUOOmejg/0r5FmXywyE5Yd6+w4UM3g/wCItkpw0cNy4x14Bb+lfG82W+8eT1r7HJ/gkeJmHxRKsjbR759ahZ/U5OamkTcoPaocbc19PHY8VjYxuc45zU/zgj0rR0ax+0qcrn0xWuNFViTt6cmsJ7nZDY5tQ1Ku4tzXRf2OPTilGjrnpWdyznihp6Ke9b7aSqt93ilTS1P8NUmMwQrdqnjt24OK3l0lQ2MV0nhXwXP4i1a1060iaSe4cRoq9ye1bwjcDgktmJ5FOa1cA4Wvr7xZ+wn418LeE5dca2iuY4YvNlghkDSIuMkkd8e1cR8G/wBm7WfjFrVxp+kpEgt1zNLM21E5x2zXRGMWroeh88Lavt54NL9lccAV9H/Gf9lnxF8FZrZtXgjls7klYru3ffGxHUZwCD7EV1fw5/Yf8WfETwuutWyW9pbypvhW4cgyfTA/nTtHl5ug1bufI32Nsfd5pgtXz04r2qH4LasfiAPCMlqU1b7T9mMLcYbPr6V7L8Sv2CvEngXwbL4gjuLe/W2TzLmCHO5FxyeeuKHFRaTe4fM+Mms2A4FJ9jc9RX0z8Bf2YdR+NeqXMFq6WtpbD99cSDhSegHrT/j5+ytqvwRv7MXTx3un3mRDdxDALDqpHY0WXNy9RpdD5hNnJnpSNZuK+4Ph7+wJrnjbwPBrb3ltYS3MfmQW8oJLDsSe2a+e9W+FOpaX4yuvDk1ox1K3maGSFBk5B5xSVpNpByo8h+yv6Uhs37Ln8K+x/iN+w/q/g34cJ4vsr+11aySJZriOHIaNTjnnqATVD9nb9kS8+N1reXgvI9O0+3bYZpFLFm9ABR7tr3Fyre58krZvj7pH4UhsnzkggfSvpL46fs16j8FfF9tpN7JHdw3aiS2uoQdsgJxjB6EGvaNJ/wCCdGrat4Hi1T+1bWHVZoBMlkyHHIyFLdj+FNqMUm3uHKu58CNZv2FQ/Y3HUYr6P+FP7Pd78SPiQ3hOMpa3MRcTvIMiMIcE47813f7Rn7FGofBvwuuvwX0WqaaJBHMyptaMnoSPSiUVF2e5LSPi+SBuOKaYWz0rrJ9NEbY2VF/Zw7JWJnY5lIXLdKnWydhwM10cOmqz428/SvoL9mv9mG6+OWoXcccqWdlagGa4cZwT0AFO6W5cUup8vx6fJ02/pUp02X+7X1R+0N+y1P8AA3WtNie5jv7K/BMMyDByMZBH4ivb/Av/AAT5tvEnw/ttRvtX+yandwedFEsQZBkZXcc1r7sYqTejL5Y9z86Dpz9lNIumyYJK4/Cvp/4e/s333iz4vXHgmRo4Li1meO5k6hFQ4JHrXqP7Rn7Edn8KvBJ8SaTqrX0EDqlzDNEEK7uAwIPrVWSkovdj5V3PhH+y5Ou3igaW5428195fs4fsW2HxU8G/2/q2oPZW0zMsEcKBmOOMkntXm/xA/ZkuvBPxos/Bcc6XRvpI/s1xtwGVzgEjsR3oVnJx6ofLHa+p8pnSZR1Q4pjaXJ0CnNfpf8QP2AdG0f4d3t/p2oyyaxZ2xndZAPLfaMsAO3evE/2X/wBmG3+NGsaiL2f7Lp9jgSsgyzMegFSuVxclshcsd76Hx3/Zb/3TSf2TIf4TX2R+1B+yza/BfWtKNhcG703UMiMycMrg8g/mK9r8F/8ABPvRNY+HdtdahqEsOs3Nv5yeWo2ISMgH1obgoqT2YuWPc/Mz+y3PG0mk/suQcbT7cV9c/BH9muP4hfFq+8L3kq28GnNJ9pkUZJCttwPxrvP2qv2P9G+Enhuy1/Q7qSe1eYW80U4G4MQSCCO3BoaSkodWK0drnwT/AGTIf4TTW0uTptNfpJ8Af2HfDvjr4c2Wva9ezJLfxl4Y7bA2LngnPWvF3/Zpjh/aEPgAXKyRfavLFzt58vGckeuKaUZNxXQn3e58gSaXJGvKn8qpzWbg9K/TT9pD9iPwp4H+FN/4g0K6ukvNORXlS4YFZVJAOOODzX5+XmliJ2+XI+lc94yV0KVvsnDtbNTPsr+hrrW0vdyF96i/s/JxtzWTkQcq1qx7Ui2rbTxXUvpuMfLQNNyCdtTzCOU+ysp6Un2Zmrqf7NJ42fpSNpZVSdtK4jlltW9DSfY33Zwa6pdMJXlaRdNIbkcfSjmA5dbNy2ApzSrYyK3IrrodMKsSuR9KSPTzJIflzTugOTayfuDR9hfHQ11s2mFeQtKNLLLnbz6UXQHHrYPuzihrBz2rrl0xjkbcfhRJpxUDCUXA5IWD4xjihbB88A11/wDZZK5ximQ6WS5yOKLgct/Z7t2NWbfSZJCFVSTXTHTDuG0c11HgnRba417Tor4+XayTosreilhk/lmuinaUrCOGHg+9WLzvIYR/3scUyDwxcXT7Y4mZvQV+yXxE+Ffw10n4FzQJpGni0jsw1vOqL5jNt4bd1Jr58/YL+Hfg7XfEniW41axtb+9tQv2WK6UMoQk5baeO1dkVek6ttEPmifnlN4ZubeTY8TK3uKl/4Q298kzeQ4j/ALxXivur9tDwn4Ps/ippY0i0t7RZCou1tQFTGRkgCvqLxx4F+HOkfAo26aZpotvsAMEiRqXLFfvbuufetJU7ci5fiM+eGrPxut/Ct1dt5cMLyv6IuabdeF7qzk8uWFo3/uspBr9Ff2C/Bfg7UfEHiO61W3tLnULbaLVLoAhUJOSAe/SuX/ba0Xwofidpw0u3t4F4Fy1uAB1Gen40/ZN1HStsVzRsfCn/AAh96tubg20gh/v7Dj86ZaeE7zUJNkFu8zf3UUk1+v3j7w58NtK/Z/S0trXTng+xr5LRhS+7b1z614Z+wHoXg9vEviObVY7V9RiRfsy3QBAQk7iM96jkfs3U5dhOUb2Pz0ufCd3ZyeVNA0cn91gQanbwLqMdmLtrSYQHkSbDtP419rfthWvhW++MVk2l28EFtvUXLW4AUjIBPH419R/FUfDjT/gL9jtLbTZbdrRRAI1XcPl6565q3RknFKOsvwJ9pC1z8fbTwje6lKIre2kmc9FRSTUV54Wu7GdoZoGjkHBRhg1+iH7BMfgnT9Z8Tz60lmNQjK/ZWugCBGc52575rzf9rC48Lax8YIbjToIUtfMAuPIUBWXd6D8apYdtyVtuoe0jsfHMngu/itRcvaSLAekhQ7fzqtD4Xur2URwQvK57IpJr9aPjRr3w0/4ULBY6TBpssbWgESxIuVO31HevCf2F5fAGg6xr114oW1F7Gg+zNdKCNvOcZqVh5OHPb5dSvaQ6HwJdaBPayFJUaN16qwwaqf2W1fUn7WF/oHiT4j3dxocUMcOSG8gAA+nSvCV0kDrXHWg6M+Vj916o5A6axPSrNrokk0gRVLMegArpk0fdIOeK9B+C9po9l8SvD0muBf7MW7jM+/ptzzmphFSdg0PMrz4earpttHc3NjcQQPysjxkKfxqbQ/hnrXiTzDpunXF55Yy3kxlsflX6RftbfETwBrXgeHStCSyn2KFXyVHHHaqv7HvxE8CfD34O6oupyWsWsvNJuWUDey/w9a7Pq8lFO2r6A5wWp+bUng+6iuzatA4uQ20xFfmz6YrQ1b4X63oNuk1/plxaRSAFXljIB+ma+pdH8TeE7z9pax1zU0hTRPtW6QsBt9QTXrH7ZXxY8JeONJh0/RGtpolACtEorX6q+a3TuZOrFW03PgHR/hhrevwyzafptxdRxDLtFGWC/XFUIfB13NdfZlgka4LbREFOSfTHrX6Ofs7fGXwN4B+AU+nTG3j1yVpBKpUB29DnFeKfB/xt4W0r9oWHxBr8cP8AZe92+5lQ/Y1P1duN7f8ABLlOKVz5Z1z4Za14aWL+09MubHzOV89CpP0qzpPwg8R65pcuo2GkXV3ZxffljiJUfjX2P+178XtA+JNxEmkiFoA37to1HGPSu28G/tDeC/Dv7PNp4fs444tVWEiZNoDFuhNX9T1UevXyM/bxtzWPzx0f4f6n4g1JdPsbGa4vGOBDGpLflU3ib4Z6x4RuBb6rYTWkx6LIpFfWP7MPxS8K+Afi1qmu6/GggkjYwMy5AbPSs39p74qaP8TvFEVzYxqbdH3ZQdRnpSWGWt9l1NOeNj5wh+C/iWbQzrK6PdNp2M+f5Z2kVB4V+FOt+NLx7TR9OmvZ1GSkSkkV94+Lv2kfC1z8GdM8P6ZDHHNDaqki7ACWxzXGfslfGjw78KZvEUupQL9ruxmFmQHjFU8ItO76GUa3MrpHxv4i+GureF9Saw1GxktroHHlsuDVzVfgn4n0fR01W70i5gsHG5ZnjIBFe8/FD4kaf4v+J9lrTWwks4rlZGXA5XcM16z8ff2ivDvjjwbFpmlQokMcKxhFAHaspYSPvK+iNVUWl0fntNppVipHfFRtp3tXaXmnpJcOxXgnNVJNNXpXht2NTmE03jpmj+zx0xXUx6airjHP0p8enxhjuHao5gscm2mjHTFN/s3jpXYPp8bZ701tOVRwvFRzBY4i6jER246VRuOI2xWxrsYhunAHSsW6b92a6Y7HM3qO8Pw+fr2lxH/lpdwr+ci1+i9v+7t417KoH6V+fHw/tjeeONAhAyWvof0YH+lfoNH90D2r5bN378Ue3gVo2SlsVG/U0pzuFMkyTXzx6nQhkPJrwD9qS6MelWkQ53Z/nXvrd6+av2qrw+fZw542Zx+NelgY81ZHNXdqbPm6TvTR0p7YbNM+7xX2B4IfjRScnpRQB7zo02y5hbHIYY/Ov0U+FN0bzwfpch6mBf5V+cOmSFJU+or9CPgVdi48BaUQc7YgMfSvkcyWiZ9Bhne56U1ItD5FNU/L714R2j6rXy7rWQe1Wc1DP80LD2poR4X4f0pJPFXjHTWPy3kTjp0DxkH6818KSMVwGGcjOa+9dFP2T4wagpbiW3Q7foT/AI18MeJrJ9K8SavZS8vb3c0P/fLsP6V9Pkk3zSj6HkZhH3YsyZf9WcHvmoc9fWp9u5SCcd6r8hs85xX2kTwJHXeBEE8s6NjoCP1rsJtNTORXE+A5D/aEi9mjP869Bl3IvAyK5K3xnXT+ErjTo9gA9KZ/ZShg34VchO5c05pWVsbeM1gbFT+y0fjFNj0tVY1on5Uz1NRRyFnwRihNjI10leO9dj8OtVfwf4q0vVo13m0mWXGOuOK51SVxxmuo8G2MWra1Y2s7eVHNKqM390E8mu2i9SXqff2pftgeC77wHel5JP7SktGQWmz7zFSMZr5h/Zy+NQ+Dvia8uLi2M+n3oxIF5KkHqP8APavrH/hkbwBqXgxLaOCQXb24K36Skndt+9jpj2r5j/Z1+Fuj+NPivdaLrjiS1shIDGjbfOZSy4z1x8v61103T5ZKK06kWjY6n9rD9oLw/wDFTwtpOjaEZJWS4+0TO6bQvy4AH510/wAAv2tNE8K+BbXRvEEU6TWY8tHhXcGXtTv2r/2evCfgvwKniPQbU6bNBOkMkCuzJKHOM8ngj/GtH9ln4F+DfFvw5TVtXs01S9mkZWWRjiMDjAA/rTcqTobe6Hu2PnH4j/E231/483HjPS0ktbb7VHImOHwqhcn3OK+hfGX7Y2gal8OdR0mK1u7q+vbF4A7jhWZdvJ/GvJvjr8ItH8D/ABwsdBspWi0m/wDKm2s3MKuSCufTj9RX1RrH7MHw/vvA81rBp0cMgtSY79W+cMFyGJ71UpU1GLkvQr3ep8nfst/Ha2+DmqahDqVu8unX2CxjGWVh3rZ/aw/aE0n4vaXpGmaLbyrBaStO8swwSxAAAqH9kj4beHvH/ji/i1tVuYrSHzIoGOPMOep9q9A/bG+Cfhjwb4PsfEGiWa6fcfalt5IYz8rqyk5x6jH61bdP26uveG1FSF+Ev7aGleG/AtjpmtWFxJdWcQjV4VyGA4H8q+dZvixI3x2m8fx2vMl603kOM/IeNv5Yr65/Zv8AgT4G1z4WafqF9YQ6peXis0ryHJQ5IwPSvnbx98MdA8MftKDwnHM0OiPdwZZiCYkkUMVz7E4oi4e0koLXqEeW71Nv4yftRya94Zk0bwrC9jpuowtHfWtxHlUJ5Plntn8qzf2Xf2j4/gzDfaXqVpJeaZcyeaPK+8jYwTjFfUXxc/Z78Ct8KNYe00uCymsbKS4gvIvv7kQsMnvnFeCfsU/D/wAKeNtS1mXXbeK+urbb5FrMflwepI71mpU5wlLl0E+Wxw37UXx2tfjR4m0ifSrSa1sdNiITz1AdnLAk/TgV7T4J/bitNL8HW9nq2i3U19b24jWWH7rELgE5rl/21/hb4Y8A3Ph3UtCsotPlv3kjntoeEIXaQ4HY/MR6V9B/Bn4S+BL34U6UE0mx1NLq2DT3E0au7OR83zdsH0pSlTVJScdBPlsj4Q+GPxgvfh98WpfF0UHnGeSQzQ/3ldskV63+0v8AtXWnxS+Hw8N6Xpk9mLmVHuZJuRheQo/HFct8PfBPhqb9pWfw3qBU6Lb6ncQxbjgPsZgik+/FfSH7Wvwj8G2XwW1XVrbS7XTb7TxG9tNboELEuq7TjrkE1VaUeaPNHXoNqN9T80rmxRmyRzVVrNF7YrTulYv8tVtp78muKW4kMhs0XkCvoL9mX4933wTuL+KHTzqNpeFWeFTg5HcV4FGr5XjjNfW37Cfhnw54i8WapJrccFxc20Km3hmxgnPJwetaU3a/Mroem7OT/aK+O1/8a9W0tpNOOm2mnq3lQscks3Un8hXr3gv9tTUtC8E21jL4da7ubS2ESTq3ykgYBPHTirv7dHhfwzo2m+Hb3Tre1s9VkuGjeO3AUvFtzkgehA5969e/Zs8KeFJ/gzosi2Vjcy3UBa7aRFZmYk5Bz7YrdygqSk4aBJQ5Uz4b8G/GLXvCnxUuPGcKR3F/dSvJNGwwrhzkivRfj1+1LrHxW8Ip4e/sYaTaTOsk7byzPjkDp0zRovh3whD+1dPpTCA+G49RdUVmHlZAyFz6buK+iv2tPD3hKH4L6hcNbWFveQGP7C8Kqr7twGBjqMZrolOPNC8dX17A+W6PnX4CftOa38LvCzaFDoi6vaRuZI/nIZc9RwDxXAePPjFr/jD4rQeMZ0W1vrV0NvCo4jC9BX1R+xLY+GLr4f3Urx2kusG5cS+cFLhP4cA9q8j/AGo7Xwxa/HrTo9PS3S3dYTfLBjZvLc9O+MZpKX71xUde41y3sbXjr9r3xR4g+H95p/8AYH2I3cBge+G4rgjBxxjJryz9nX4va/8ACnVr1dJ0/wDtNL3HmW+DkkdxivvTxXpvgqb4X3sF2unx6Oti2MFQFATgj36V8wfsOyeH/wDhKtdOoG3W9WJfspnx03HcRnv0qITXs5NQ0XTuHuWZ5T+0b8W/EfxQ8QWR1awbSorIfuLVlIwT1Jz616h4b/a48cad4DjtY9CFybeAQrehCQuFwCfeui/brvPDjQ+HRam2bWPNfzWhwWEWBjOPevcPhHdeCU+EukLA+niz+xKblZNuS23592ffNJzXs4ycLrsTeNkfBHwn+J3iHwT8SpvEVjB9tvrpnEsJBPmbjk/rXaftG/Gjxj8SNHsdM1nSDo1pG/nCPYy+YwGM8/Wtf9n/AFPwvH+0hfSzeTFpPnXJsfOxsB3fuzz7V7X+2nrfhq6+GMKefa3GqG6jNt5bAuF53dO2K3qT/eRi4a9+w/d5jxD4LftD+O/Bvg1dG0vSP7TtLRD5R8sts59q8qPxA8TSfFj/AIS7Lf279p84ALjDdNuPpxX25+yXrXhW0+EGnmK5tIb/AC320yMquW3HrntjFeC6rr3hJf2vI7qP7P8A2J9uUu4AEXmbeW9MbqIzbnJKG34itBXsZPx++OPxL8XeBf7L13RTo+j3RUvItu6edjBAJb39K+R7u1XccjJr9Sf2sPG3hG5+CWs282oWd5POqraRxSK7eZkYIx0wM1+X15CZGY9MmvPqy5op8vKSrNXRkNCobpxUbQJnoKttGV4qE25BzmuFgQfZ09OKQwquflFWGt92ADik8gouM5FSBX8pfQc0nkL3AqcWp3Zz+FJJDu4BoEQ+Sp7fSk8lPbIqb7OUxzkUjW7O2S1AWI4YV3sBTgqKcKFz3pyxlSQKbHbFATnmgQjqq8nGKTav4U5rcyYxxinfZyvGaYEY2k8UkmwKM09LcIST1pZLUSKMnvQBF5ankdKFWM8DrUnlBAVBojtgvPegB0cYRs1u6DZPf3sMMKlpZGCoqjkknAFYyweZ1NdV4N1BvD+vafqEKB5LWdJgrdMqwYD9K7cP8aEz6h8Ufs//ABa0X4WfbL7Unl0yGAO9j55Z4kx3H9K4b9nv4Q+M/Hmr3b+FbxtMa1XbPd+YUUZ6KSOua99+IH7Z2jeK/hvPY2dpNb3t3B5cisOAccgH615f+y3+0TafBv8Atu01OyluLS9ZZVeIfMHAxj6YP6V7y+s+y5nbm6Ix5ovoec/GD4Z+KPBvjhdL8QyG81KbAilDlxIDwMZr07xP+zH8StB+GY1G71VpLSGISNp4mZjGpHTB44rj/jf8aD8SviHbeIbS3aH7I6GJX/2WBA/SvY/HH7aNt4s8AnTY9Nls7yaEJKc5G7GDj0FWvrDlGz/xCvFdDxr9n/4I+KviXqt3LoF62lx2g2zXZcrhj0XjrWF8avhb4g8C+Mv7L124N/dS4EUoYneD0/GvQP2bv2kF+DLava3lgbyy1CRZgycFH6fliuP+Ofxkf4neNIdYhQxi3w0Qb6g/0qeWq5Su/c6F3XRHX+JP2WviBoPwzGr3d/m0jiEjWHmElFxXJ/s9/ArxB8VtavTot9/ZsNmn728yRgnovHc16Z42/bKu/F/w7XRpNNe2umhEc0qn5WIGM1wf7OX7Q158FZNWgWx+2WeoAPtHVHUHB/Lio5a/Ik5e90Jcl/Kcb8YPhZrPgXxwNH1Ob7bdO21JQ2d2TivT/GX7JPjDwr8MBrd3qilEjDtYmQnYCOlea/Fn4qXnxA8YJrJiMUsMokjDdQQQRXp3xE/a+1fxx4Fh0aXTmtGaERvKp+VjjBNW4z5rRltv/wAAmLutYnGfs8/s4a38Zrm/m06/TTLWywst0xPLHooArivix8K9W8A+PW8P3sn2y6d1jjkV8hsnAru/2fP2jNY+C8Oq2drZLe218RIR3RwMZ/LH5Vw3xN+Jeo+O/GI12ZTDcxuJE56MGzUcsmneXu9Bc/vWUT0n4j/sh+Jfh/8ADeLX7/VYpB5Yd7FWOUGAcc8E81y/7Pf7MmsfHD+0JrS9j0+ytDte4fOd2OAAK3fiR+1R4n+IXg2DStStdirF5ZmUYD8dfxrG+B37SGt/B3RNY0vTrQXKX0gffzuQ4xUyjJJRcve/AvmdvhPKvi98Pbr4b+LLvSbuZbhoThZV6MPWvP8AG7PGK7z4k+Lrzxx4gn1G9XErE5HvXHMq9MV5OIUVUai7lRvbUqK20gYrt/hZ4Fu/iL4y0vQbLAuLyYRBm7dyfyrkY4wpB712Pw18Wal4H8XaZrWk8XtnKHjyOD2xUUfiSK16Hvn7Qn7K5+Deg2l2urm+kfHmRntk449q3vgj+x7afET4Vz+LNS1RrRJEc28UfouRkmvM/jN8avFvxFVBrsT24YjAZSqn6Vo+APjh498OfDk+HNNtZZtIjLuZEQnaGySCR25Nex7P4Y82vVkNy7HH+Gfg7P4o+K8Pg6C6XfLPs849NvHP5Gu4/aM/Zyg+DYtha6k16cDzFYdM+leZ+H/G2u6P44ttcsiw1WKQlMLkk+lbnxT+InijxlMp8QxzRz4581SuR9DWrVN310MZe05lZaHrXw1/Y/sPE3wdbxjqmqyWrzxl4IYh+Wa8u+DnwMf4p/FNvCy3iwQw7mmuAOQg9vU1Z0H4yeO7X4fJoNp50ui2wZUdYjhR6Fh9a5b4d+NPE/hjxgdV8N+Y+rTBoykaFi4PUY/WseWHKlzav8Dp5pdjsv2jPgXpvwh15LLTNRN5H91t4Ge/PH0r0Ob9j7RNJ+Ctr4n1PVXGp3cAmSNCNqgjIGK8I+J3iXxP4g1bPiMTQ3jfMVlQr+Q9K3NS8dfETUfBdvFefbptEt41jSZkYIFAwMdq0UaV99t/Mzcp22Mr4U/BuT4jeJrm0+0JDY2bjzn/AIipOOKsfFH4RWXg/wAc2ujadeNPBcSrEGfBK5OP61h+ANW8UaXqlxJ4cWeaeZcTJEpbI65NU/FWsa7qfiISal50V/Gw2qykMrZyOPrRzUPZaJ3H719j6Z+Mf7JvhX4c/DW1v4dSeXVpIQ7ksME4zjFcV+zH+zfo3xY03WtV1rUHtbWyG1VjbBYgGuV8fat8Srrw/by+I4rtLBoxskmBHGP8KyPhrf8AjvT9Nv4vDEd1NZSHM/lKSAanlhom3fuElPoizdfCWxuPjRY+E7S7xZXVx5QlY8gA8/yr079pz9nPwh8K/D8TaNcvNfKgZ23Z3H3rwGO68RyeLYpoRONcSbESqD5gfsK6n4rL8QLS1iPi2K4jEw4aXv3xWM/ZWkxqUtEeIXEEjSEDmq8kJXoOa2ZNq5IPNQFV6tXzstzczRGQPmFEcDyMT1Aq7Io4PUVHG2GwDj8azArsp3ACpNvy4PWpGVVXk809bfdHuPSoGed+I9rahNjgZ4rnrrHl++a29ecNeS+zGsO7+6O3Ndkdjj+0dj8D7X7X8VPDy4yFmZz+EbGvuaJuMivij9naHzPipprdRHHM/wD44R/WvtKGTgV8lm2tZLyPoMF8DLYb86ZJnNIr85NNdvmrwj0SJsjNfKf7Ul1u8SRR5+6i5/KvquRvlOPxr49/aUuPO8cSrn7oHFevlv8AGOLFfwzxxvypp9Kc1Iflr6w8QbRRtNFAHtOnn5hnPHIr71/ZrvPtHgGz/wBkkD6V8DWZKmvt/wDZTvPO8GmMtnZIa+YzKN6Vz3MM/ePfX5pI+maG+YULXzJ6ItRv8yt9KkJqNuhqhHh9632f41W4AwZLVhn/AIEP/r18b/GazFj8UPE0YGFN/M3/AH05b+tfYvjBhZ/GHRH/AOequua+UP2hLXyfi94oR8HFwrDH+1Gp/rX0uS6VX6Hl5h/DXqebM3Q5yaa2Mgg4pJGA/GkB6DvX20T55nS+B2C6sM8jaR9a9PjC+X2ry3wWP+JxEp4Ocfoa9TW3ZlAWuWt8R00dhBGFHyjin7F70kULL940SwuzAqeK5joAqN3Til2rvBpyRfLg9aZHbusmT0poCzHGpbPatTTSyXCNGSrg5BXrmsxYywwDitXRMW9yjPllDDI9q66L94R9TeFPi18YbPwKq6fptxdaasO1bs25bauMZBxXiPhXWNd0nxZFeaTJMNYaQlfLBLsxPp35r9APhH8XvCOoeANNj/tK0s2t7cJLBKwTGBg8V8oeH/HXh/w7+0pfazGiNorX8jRlVG0ZPUe2c16NKUryXLb9SU1roVPjV4s+KWteHbOz8Z2t3BpquJEaS28sO2OMnuap/BPxD8SNJWWHwVFc3EWd0iRx7kH1/OvqX9on4keEdf8AgvrUcGp2d7cXEQ+zRBwX8zIIIHtXnn7HnxV8O+GdBvtH1W5isrqSbzFmk43e2fxqozl7Nvl+RN1a9j5z+KGr+Kda8aTXvilZotbGxSJEKFcdAK9XtfEfxrl+HMzW8V4+iNAwMuzLeWRgkd8Yp37Y/jTQfEfj7RW0maG7a1g/0iaHocsCAT7AH86+j/AHx88FXvgGyafU7e1eG1VJLeQ4OQvIx+FOVSXJFqPyC67HwT8NZPElh4utv+EbaZdVZtiCHOW9jXoP7QV18U5tM0238cpKmn78w8fKW9yO+DTvhJ8TtF8FfHDUNduI1GlT3M/lYXhFZjtP5GvZ/wBrD4weEPGHwlNjpt9Fe3808ckaLy0eDkn8v51rKcvaRtHQvS+x4t8EP+FqzaXMPBTXEllCfm/uA+gzXmfjZfEknjq+k8QfaI9eaYeaZAQ+4cDH6Yr6f/ZL+Ovhvwn4RfQdZuEsHSQyLM/R815l+0d8StD8S/Gu01nR/Lu7S1WFZJVHEhU5/Hg4o5p+0acdO4la+xreJNA+Ng+GMhvWvpfD72+ZYwQz+V15HXGK8o+Dmk+L9S8Ui38INOuqEZ/ctjA9Se1fcbftQeArzwLNNJqUUc5tSps2Xkttxtr5Y/Zb+Lek/DLx1fXGqp5dnex+WJQMmPnIrKnKpaTas0Catsc78ffD/wARrHUrA+PTO52FLaR23RkcZCkHGenvXW/B/wAD/F7U/Bclz4Uurq30pshB5wTee+0E9K7H9sf40eGPH3hvRNN0G8TUJo7kzySIpHljaQBz65/Suq/Z7/ah8JeHPh3p+ia3K1jcWSFNwQlWHrV81X2ako69gb93Y+Rrfwzr3/CcHTUjuU8RfaihTJE3nE889c5r1n40fDX4taX4F+0eJbi4u9EhCtIiz7xHjoWFZWsfGSzX9oubxxZ22bAXqyBMcsoULu/HGa+gvjR+1H4M8SfCLWbCymabUNQtjCtuy/dLDGT9KVSVSystOoPfY/PK5wvFVeMnvV+6j82Rie5zVZoQvSvNluAkJG4CvU/gl4N8ReMvFCWXhmSWG9YZaSNtm1e5Jry+OEBgT1r3j9lz4vWXwh8XXF9f2z3FvPB5TeWMkc5rWje/u7lxZc+Ovwf8Z+A3sr3xPcSX8EreXFdNKZFzjO3PbgV1Xwj+AvxA8aeDzf6Pqkmm6c+RErztGJfXAHbtWn+01+0lpfxY8NWOj6RZSpGk4uJJpsA5AIAA/E11XwR/a10XwZ8P9P0PVNPuN9ihRZIcENyT/Wu1yrcit8Q+b3dj5uk+Huu2/wAQn8PSRyDW1ufJKhju3565985zXrHxc/Zn8ceEfBH9tajq66vZWoDTW6ysxgHrgjBA9q5K8+M0dz8en8dpat5Bu1lWHvsAwB9cCvY/jJ+2FpPjL4c6noek6Zcx3eoR+RI82NqKeuMdeKUvbe7bbqJy1VkeU/AT4H+J/iUt3caNqJ0q0gbbJcM7AFvQAc1z3xi+E+u/Dfxmun63OL2e6AkhulYt5ozjOTzkV6L+zT+0bafCXTNQ0zUrCW5tbiQSq8PVWxg/hXMftCfGxfix4407UrWze1srGPZCknLH5gST+VaR9q6ln8I1LXbQ7p/2UfHF98P/AO0n1jcxt/P/ALOeViSuM49M47V5P8FfhXrXxI8WvpukTmymiQvNNuK+Wo45x719Dn9t+0i8H/ZRoUx1Jbbyg7MBGW24zXhfwG+NUvwr8ZX+rvZm7jvI2SWJeDy27j8aIOvaXNv0Jvo9Cz8ffgR4h+F81jeatqA1S3uyY0uAxJDDnac+1dv8Of2UPE3jDwHDqkesDToryLfDbs7fOO2cdM1zn7RX7RE3xgtdMsodPaxtLV/NYMeS+MV3/wANv2w28K+A7DR7nRZLqezh8pJkOAcfdzUv26jo/e6iu7bHhXg/4T6rrnxM/wCETj/c6mk7wOScBCp+Y59MA16R8df2XtY+HfhdddfVhqlrCyxy7id0e44BAPbOK4bwf8XL/Q/jBL40+z+bPNcySyQ+oc8j8q9F+Pn7T118S/CQ0K30ttPt5ZFeZpDktg5A+ma0l7Xmjyv3epd3fRGf8Df2YtY+JXheTVYtV/suyZzHHy3zsOpwP51wHiL4M6r4f+LA8HSyrJqEk6JHID8r7+jV6X8D/wBqa8+Gfg0aFJpH9oRQuzxyBsYyen515p4x+L+q+J/ixH4yEIhvIZUeKIDIUIeBRH2vM3J6dCbu+x6r8Yf2N77wf8PLnXLfWxqMtjF5txbMhUKvfYc9vwr4zvhsYqPWvsn4pftea54x+Ht7oh0VbCS8i8qa4HIIPXFfH13H82T+tedWVS16ruyLNLVWMllzzioGbL4xV5xz7VF5agYAzXAySqxCDIGaEbcvI/Cp9oHUZpGUHPakBV8z58U9uBnv2qdY1xjFO8nnnpQBVHK9KjaQ7sYq8YxzimrGozxTsBW2hSSOTio4ZDIT8uBVrZmTHapVjVRwB6cUWAz5iVYYGc+1PGduSCOKuMo9qTy1b6UWJKC7pG5XAomZkC7RmrTAdFNI2FHPFICssbbcnrTI97McjirvVfWkUqcgYpoCHLLgCu4+G+kwa54m0uwvJRbw3NzHC8rcbVZgCf1rjlYRtzXQeHYpLq8jSAkysQEC9c9sV2UPiJZ+kPxk+Afw68O/CW8+z6bBaXNvbg290HO9mA6k55zXjn7Ffwy8I+Nh4hvvEkFvf3FmY44redvlUEEs+M89MVzPjzwX8YIPhvb33iGS6uNESIEJI+50TH8Q6j8a434IeAfHXizUro+C/MjljT99MH2IB2BJ459K9iEYex5HU369iLy3Om/aC8GeF9D+M0Gn6IqWmmTyRrKqH5EywBx+FfSnxp+Enw90P4QTJaaZZwTRQL9nmjP7xjjrnvn+tfFPj7wv4nsfGh0nXopv7W3hAGOSxJwMHvzXp/xC+EXxW8M+AYr/AFi6mk0uJFzCLgu8S44BFaTjT5oPm0X4jXO1odT+xb8OfBni5fEF5r0UF9dWjosENwwwEwSXx3rzr4/+GfC+m/GKO20fy7fTJp0WVYz8qZYAkfrWd8E/g741+IE17ceFna0jgULLO0hRc9l9zXMeOPBPiHR/Gz6LqyNNqzOIgA27cxIxg/WlH2cZSnza9uw25bH2F8e/Avw28PfBsw6faWfnrEpgmhI3nA+8SOteZ/sOeGvBusR+Ir3xBHa3F7blFgjuiMCMgkkA9TmuI+In7PvxD8F+BI9U1e4aWwVQXg80sYwegIrnfgd8D/FvxMa9uvDkptI7cbZLhpCik/3eOprL9z7PkUn5sj3+u5Y+O1n4Zg+NqxaeqWulS3KCYRfdRS+GP5fyr6I/aP074aw/CWK10e3sftEca+TLbY3AAdSR3r5C8XeBNZ07x22hXwM+qPKIh82d7E4GK9F+KH7M/jP4b+CYtX1O7Wa025khjkJKAjofpWz9jzpu+my7ijz9z0j9iOz8DSeH/El74ggs5tTilVUW6AOItgOVB75zXi3xLt/DT/GxPs0Udvocl3H50ceAApf5vpxWj8Df2d/FHxasdQ1DRrxLC1tTsaR5Cu9sZ24HWuH1X4d63afEFfC06+dqss4hX5s7yTgEGpjKkuZrVu1/IXLO+r0Pq79qub4dN8M7aw0O2sPMijXyHtVUMOwORzXMfsdw/D7Tfh7ruqa+tnLq3mtGq3ABYLtOMA1538bP2Y/Evwo8M2up6lqcF5E4Ae3icnYeOOevWmfBX9lPxB8WvCt54httRj0uyjYpHuJzKyjkcelZydFQiteXv1KXP3PH/jnNpt5471GXSUjitZJCwWPgfl2rzho9vJrrfH3h258J6/eaddusk0EpRmU5z6GuUZs9q8uu1Ko7KyL23I0jLP1xXr37NeoeHtF+LWg3XihY20mOYmTzBlQdpwT+NeSRv+8xivUvgh8J7/4weM7XQrGRYnYF3kboijqaVK19dha9D6C/a/8AiB4T8YpDHoMduY4AMSxAAE/hXY/CX4reBvDn7N1rpzG2OsyRMtwrKN5bPUmvCPj/APs+y/Bm8t4jqi6jG4Ct2Kn6V3/h39jk3vwfTxbe6z9mnuIPPit1HG0jjPvXs81P3E46aW8ybSafvHB/ATxR4V0v48DVfEKQppA8xoxIAUD8YP8AOtj9rL4ieH/G2tq+kCExq2FMSgADHt2rivg78D7r4qfEKTw3DdLbJAGaSY84UHGRUvx4+CafCfX0sINSF+hOzLcHPpVOSvJ8upXK7bnsuj/GrwRpfwE07QbSC3TVBb7Z/kG4v3z+ded/srfEvwp8O/iVqmq67HH5LWz/AGUyKCA2e3HHFdNqX7HtvoPwltfE97ruzULmASi2AG1c84rz39n39nmb41eJNQsjfLZWtiu6abgkdelYyqRlFe7p1IjTf82o79oz4k6T8QPFy39gqNAGLB0A+7/9evWfib+0F4R1T4WadoelwQoY7NVkVVAO7bzn1rwv4qfCOLwT46h0Cx1AXsc8ywpIwwckgc4+terfFz9kfR/hv4Dt9T/tprjVHiWRo8jbyM4xW3tHzfBr0I9n3kc7+y78WvDPw3k1+XWYk+1XCf6OzLnrn8q4Xxx4/wBO1r4nWutrEslrHcrK6qB8ygjit74K/s/n4naPqGsz362+n2Umx0Uje3HasOz+E9rqnxe07wnb3oNtdXAjEzdVHU5/Kjmn7O/KUklK9z1n9o39obRPiD4fistMQeUsYQKowOlUfgD8fPD3w7+GOq6Vc24/tWeRtr46qenNWf2mv2efCnwq0WH+x7xpLtFG7c+d3rxUPwj/AGc/DniP4UXXirW77bMSRDAj4xjv704ylJq0dOhpK0t2eY+D/ibp+j/Gmw8T6hAJbSOfeykcd+f5V1H7THx5s/iplbQDy88Y/wAKzvgv8H9I+IPxk/4R29vVi0yIs5Zjgso7Zq7+1R8KfCvw/wBSWHw5JhUbDjdu3DpWM5zUZJx1I9km1O583OgzURjDd6kYMWJxxTW3dF6mvmZbnSDIrYA6UxYUXccc1LtZeD1pERj1rMBqwg8tUioAOKUKzHAGBT3UxoSTt2jP6VIzyjXsNdOVAwW7VgXRPANbGrNm4PGOc1j3f3l+ldvQ41ueofs1RbviNv8A7tnKf1Wvr23b5a+Uf2X4Q3jDUZe8dptH4sP8K+p7dv8A9VfHZlrXPosHpTNGM0jHrTUbNDfpXjHcyNuFr4r+Plwbnx9fnP3WIP519nzPtjb6Zr4c+LV19r8ZX7nqzt/M17WWL32zhxWkDhc9eKaf0p7Limt6V9OeMN3e9FH4Zopgey2P3uuK+wf2Rb7dpN7BnhXBr49swNwNfVX7I9ztlvo8+hr5zMVeiz2sP8R9ZfwikWiNsxihe9fKnpi0xhTjTW4oQjwz4qKbX4i+GrnH/LVkz6Zr5p/ap00WPxev5lOftVtBMR+BX/2Wvp743QldY0K4Xgx3Qya+e/2wIRH400ecIoM+nr8/c7SeP1r6HK52xEEuzPPxq/dM+fp+DTEbdg55qSb5o8g5xVdFNfeo+ZZ0XhFtuuW+Pl+cDNexw7C6B2ZY8/MyjJArxLQZGTUrd/7rqePrXt0IG0EHtniuTEbo6aOw1ZA2cHdzR5ix9aVYQmQO9OMQYciuY6QVgxBFJ5g8wr39acqinmFRjimIT7vPWr2nyCRhjk1UAVuKntXFu+VraD11GfXHwn/ZFufHXgmHWbnW/sBul3QxIm/jtnkV4r4g+H934W+Iz+FdQdUmjuVhaVMkbSeGHtg5r0j4Rftd33w58Mx6VNaC9giGE3Hp+teS+KPidf8Aibx/deKJSFuppvNA7D0H5V6kZPm956GfMz6N+In7Ha6L4Bl13Rtba/uLaD7Q8MibUdcZO0g9frXnn7PPwJPxiur15b42FpagbnVcsSfTtVXUv2rvEz+HbzR4Cq2d5AYZEb5tuRglfSud+Dnx01T4S30k9ntkhl+/Ex4NaXavHnu+jJ5nbY3/ANoT4HzfB3XLOMXhvrG8BaGZhh8jqCPxr2P4cfsbad4m8CWmpahqk0N7eReaiw/cTPTPrXgvxy+P1/8AGi6017q2itI7LdtSIk5J712Hw5/bK17wL4cg0h7aK/ihXZG0nUDtWT5+WylqPndlY5TQ/g7c33xkfwTNOoeG6aJ5h02jv+VeyfG/9kbTvBXgG713R76aaexTfPHNjDL3I+lfNc3xW1b/AIWFceLI5TFqEtwZyVPTPavRPiB+114l8deDp/D86RRRXChJZF+8y9xVOUm04yNOd6JHYfs0fs46T8VtDvdT1e6ljhhkEaxQkZJx1Ncf+0F8EIPhL420/TrO6NxZagu+FpOGTkAg/nXPfB39ofW/hH58Vltnt5DlopDwTVL4s/GjVPjD4ktdUvlS3e2TZFHH0HOc1XNJz5ubQXNLmPq3Tf2LfDWpeA4ZH1C5GrTW4lE6kGMMVzjbjpXzn8Gfg2vjz4qXPhm8uRDFZySLO69WCEggfXFdz4U/aN+I+l+Cfs1rp0l7ZwxGMXflM2wY9cV454H8aeIPD3jQavpLt/askjMdoLFyxyRj8aqMZ680vQqLlrc9v/ae/Zn0X4YeG7bX9BnmEHmrDNbzkMcnowNdT+zd+zX4V8afD1NZ1oSXc9yWVY45NojA47d68m+OPxY+IXjLRrTTfE2nPptiHEq7oGTzGHQ5OKPgj8QPiP4fsLi08KRXN9br87wxwmVR74quWo4crnr3D3uUzviF8HrXwn8bh4Qguy9pNNH5Ur4yqv2PuK+h/il+yT4Q074Y6je6f5sOpWNqZluGfIcqMkEehr5M8Z694h1jxxc6pq5mTWjKC3mKVZSOgweRivUPFnxH+Ll78N5bXU4LoaJPEEe5aAjKYxy1ZTjJpJS2H79tD5duzhsrVVdzckVenQRsQexqv97OK4pbmYwA7hxkV7x+yj4D0n4gfESDTtXG61WNpGTON+B92vDFYKcV2Xw5vNbsfEdvJ4feaPUWO2Pyc7j7VpT1dr2Gr9D66/aw+A/hLwf4Dj1zRbZdNu4Z1jMatxIrcHj1HWtj9l/4H+D/ABF8OItV1ayh1O6uXYHzCf3YHGOO9eA/GSP4oz6baN4zF8LDP7oyrhN34d6t/Bvw18S9c0meLwlcXkdmh+fy5NiZ9M10ci5eVz+Zdp8tg8efDXRtL/aEbwvYSiPTJbmNeWz5Ybkrn2r6V+MX7PPgPTfhTqlxb2EenXVhamWK8VyGZlHAbPXJ/nXxb4m0fxDoXjaaDVPtEWuxygs0ufM3dj716h428K/GC6+H5u9abUZ9BVRI6vJkhfVlBzj61copqL57W/Edp6WOw/Yz+Gvhvxeur3+sW0V/cWzIsUEwyoBzliO9Yn7X3gLw/wCD/HGjDR7eKyF7GTNbxcKpBABx2615/wDBPwx408QavLD4QnuLecLmSSOQooHuazvjd4Y8Y+GfESDxe80l5ImY5nfeGUejVdoqrzOXyM3zxlc+2fDvwJ8Br8L4YrnTrWfzbPzJbx8b9xXJO7tivmf9l7wb4c8Q/Fu+stSEd1a2vmGCOXG2Qg4GfXivFm+L3iiHS/7MXWboWm3YU804x0xXPaT4kv8AQ70XtjdyW1yORIjYNQoqN1z7/gQpy1Psr9tDwD4W8O6DpWqadaW9jqDz+U0cGBvTB5Ir0T9nT4X+EZPhTYXd3Y2l7c3kZeeSYBiPb24r8/8AWvG2seLZozqmozXrJ90SOTj6V7r8Jfhb8SfFnhF7nQby4ttOwQoacor+oFS4x9nyufzNFzNWuWvCvhfwq37TUujsY20JLxhGjN8pIGQufTPFe5ftaeGfCNn8KLieO1sbXUY3QWrQqquTnkcdRivj7S/BHiGbx/8A2IkUy679oMe3JD78+tek/GP4BeN/CPhqLVtYvTqVlHjzAspfyifUGtZRhKUW5/LuN811qew/sb+EfC+o/D+8vru0tLu/knKSfaFViqgcAZ6d68d+Jmj+FdL/AGkItPtlhTRGuIfPjQjYpJ+YfT/GqfwT+C/jPx7ptzdaFetp1ojbTK0pRWb0461xfiz4a6/4f+IR0DUEaTWHkUKwbO/d0IPvTUIKo3zX8uw0pXep9r/tAab4LT4L6xG0OnxpHbE2nkhAyuB8uMfhX5falGzynHTJxX138Rv2VfGOg/D+TWLjVUvFt4/Nns/Ncsi9+vBxXyXqH7mRlPXNcNSMIxtCVyLab3MhkK8VB5J3E5q23znrUPmDJXvXE0ZkTRluOlJ5ZXgnNTs235iKRG8wZxSsBHHCWOc1N9lZ1wKkjxuAxXuX7N/wTT4xeKX095/s0EMfmyyYydvoPeqUepcUnueIJp8idqj/ALJmdiwHFfVP7Q37Ntr8JbjTZLO7N1aXreWN4wyt6V634L/Yj0nUPBdtdanqMkOo3EIl2xoCiZGQD610ckVFSfU0aiuuh+fP2Jt5BGDTV02WNtxHBFew2/wzW4+KTeFI5VeT7YbUSEcfexmvfvjj+x5o/gP4fnWtN1SSW4tQvnRzoAHzwduOn45rSVHlaj1Zk7dz4gk02RwABx1pslq0S4xz619l/svfsy6D8VNEv9X1qaQW8MnkRw25CtuxncSa8Z/aU+Ftl8J/HlzpNjcm5tdoePcBuAPY1lKnZtLdE6dzw6O1MeWzzSSW5mYZOMVYMhkbGMCoppDG2FGa5yRoj8sYHIxUaQhOal3HbnFRRyFicjFIB5i3EZrpvBOtp4b8QafflPNFtMkpU99pzXMPIy42ipkYqoPet6cuR3Ikr6H3x8Yv2xtA8YfDmXT9JVlubqELLHIPunHT864H9lH9pDR/hFpms6drEDlLpllikjGfmAxg+1fJEV1K2QRxT2vJFxt616SxFHkVPk90wjTkru5758VfjRF4t+LFv4mt0ytrMkqK3fawI/lXsPxj/bE0vx14EbSrKCS2muIwJASDzjkfnXxOt2WGT+VRi6lbBPSm8VBy5uXbbyNEmtLn1Z+zR+1HYfCDQtV0rUbQypcSedFIp5DYxg+1ec+PvjQ/ib4pQeKoY+bedJkVu+1s14pcXEhYEdKGmdUxnnFZrEpJ2jq9xcrve59gfGz9sGD4leC/7MtbdraR1AlXsT3rkv2e/wBqGT4O6BqujyWomgun81JM8q2MV80Q3EmDuPFOYyM42nimsRHRcqsuhMoOStc9X8RfFK51r4jQeJ4UxNFOsyq3qGyK9T+Nn7U198T/AA1Hpslk1puUAr/CfU14x8F7HTNT+ImhWmtOqafJcoszP0256V9g/td+H/AOl+BYRpNnZQ38eBG1qADjHAOOtdsa8qs1aN2/wEoxgkmzw/4FftH6n8H/AA3qmjxWQuba6fzRIudyHGDXnWo/EzU5/iJD4ot02XkVws8auc8g5ANfV/7KGh/Dyb4T397rkVjc6kZXWUXQBZFA4AzXhGhWPhG8/aItbe6EUXht73LKx+TGemfSnGUlGSUbd/M0ahvcX4y/tG618VrFbe/tltgBjap4/Cl+GP7UWvfDn4ezeFbW2RrbczLMM7hnqK9c/bNbwI2hWMWh2mnx3UPSS0ULx6cVe+Acnww0X4GteajHYz65KsglWdQ0gPbHoMU+eclFuHorbEL2aT1Phvxf4gn8Sazc390P3srZOea51lyOmK6/4mNYXHii/bTQI7ZpSVVRgAZrkSm3knNePiL+0lzPU1VmtBkeI5Oldx8LPipqvwl8Uw65pLhLhFKEHoVPUVwywszlt2BQyluM1jGbi7oLXPT/AInfGzV/ilefaNQb5ydx5rfj/ah8UL4HtPDDzs1jbx+WvPO3sK8T/wBWAM01Vzkk812/XqnNzEciSsj0H4e/GDW/hv4mm1vTJylzMjI/uCarePfihqXxB1L7bfSHzc7s5zzXCv8AN1OPWk9s1P1qpa19yuVHqWq/tBeLNY8OwaRdXxmt4UES/QDisv4f/GHXvhrPeyaRceT9sXZMOufeuBWLjcDz6UeWHfnge1L63Vum3sPlR1PiHx5qPiPWl1S5lP2pX3q6nJBByDW74o+NXinxnax2+o6g8yKoXOewGK868vHGeKmtxs6c1UcVVbeu5LiuqPTfh3q3jex0u/g8NQ3c1jLzMIUJHT1rAs7vXl8XxXVsJ/7bWYeWFB3huwAr6L+AP7QHhf4e/CO90S5gQ6vM7HcUySD715d4M+JenaD8ZLbxReWyy2SSliu3IHocV6i5nTV5ehmuW+xmfFJvHW6M+LY7qNpBlTMDzmneF7T4gSeC5f7Mhu30KMseAdv4V2/7Rnx2svijeZto8x/wgcY9619D/aX0vSfhDb+GoIFhnjQq2RjJ+tOKfPrP1NPd3seJ/D/SfFWr+L418OC5OsncP3Wd3PXNO+LWieLfDuqC28VrKLhucyHOa9F/Zz+OGnfCfxvqmsahbCQXMTCNsZ2sea5b9oL4sJ8Utdku40wrEnntXDWl+7fvaGit2PGpG9OlQs+1vQmrPlqq47+tQ7NxGRxXgM1E3dDT48P34pWhDdOBU0cSqoAGD9etQwGKQrYHNNvm22U7Ec+W38qtJDnkiqutAppVzt5Ow0luJ7HjupnM9ZF4xaQfTFauo5+0Nnism6/1hrs6HJE9w/Zatz9u1y4xwqRJ+ZY/0r6UtznFfPf7LsLLp2sy44aZFB+gP+NfQVscCvise715H0uG0po0Y6Gb8qbG3FNZvwrzDrK19J5dlO+cbUJ/Svg/x5MZvE98/wD00P8AOvubxBMIdFvX/uxMf0r4N8TS+brV42c5kNe7la1kzzsY9EjIwWqM/SpC2OOlRtmvokeSHFFJRTA9fs/vAnrX0j+yjeGPxFPET95K+bbTjFe8fsx33keNIk/vKQa8THRvRkexQ+NH3ND/AKsUopls26FTT1+9Xxi2PWY6mN3pxprUyTx34+R+Xp1lOP4LhD+teCftfQll8J3X961dT69Qa+ifj3Du8LNJjJSRW/WvCv2qLF7zwL4PvFG4KXRz6DaDXsZbJ/WIHLitaTPlp18tv/r1Ev3j61NJFtxiofutzX6HE+UZf0qQx3kZXkZ5zXt1u5a2Qgc7RgV4fpW1ryPcQATXt2lSD7HCzcfIKxrbJm9HqKhl8zDVNN5m392MmntIjtnPSpFlEak9K5DoIbfcy5PFObzN3+zUkZDcrTmlXcQaBjG+7kdabF5nO5am4pnmDoDTuGo5meRcJ1pI2cDnrSCYR53HFIJFbJBrXmYhXZ+oORTvMfaQDzTPtCKdpOKXeF+anzCGRl425NDF2bg4ApPOEjYFDSeXyelVzAODFevWoWEiyA5zml80ScjpTPtCs23vT5mBPy/erenN5M6FzkbufpVISlOvSrdlIJGFbQlqNH6cfBTx54Km+Fum20dzZWyR222eGUqpY45yD1r5L+Hvijw34d/aGm1CSKNdG+2yeUdvyICeD9K6z4S/sn6n4+8BR6wdaXTjcKTBFtJ3fXHSvILX4W6lN8TD4PcouorcmBmBO3g8n6V201Tu7P1Go3b1Prf9rbxr4T174RPHb39pe3rzRtaiNwzqc8n1HFYH7H/xS8K+HvBtzpmpXUGn33nGRpJeN4+tcT8Xf2Rb7wF4Il1221hdRFooa4gZCuAepU5Oax/2d/2c1+Lml3mo3OpNp9pC/ljyk3Ox/PipSpezav7oraWvoM/aO8beHtW+NlrqmmNFd2MPlefJGPlkIPP14r6b1/42eAtU+F19nUbVkaxZBaNjdnbgDH1r48+OHwXl+FPjeDSBeC9trpBJBMRgkE4wRk4Oa9jt/wBisX3w9GoNq5XVWtvPWIJ+76ZxmlJUnFNv0KcdFqfFmpYmuJGQYUscD2qgqFGJrT1m3awvLiB/vxOUb6g4rMVy3UVzT3JDytzZr1b9n3xtY/D74h6bq+ow+dbQtzxnGe9eVl9rDivRvgv4NHxB8babopk8kXEgDN6L3NVTtfUEfV37TXx+8JeMvhi+k6Tci8vrl0YKB/q8HJOai/Zf+PXhbwd4FbS9auRYzLKzhyvDioPjl+yroHgj4dXet6Vd3BurJVZ0mIKyDODj0rC/Zl+AujfFHQ7zUtYmmWCGTykht2AJPckkGt17F032Fpy2ucf8afi1pPjD402mv6dH5mn2bRpuI/1m05Jr3rxx+1l4N1L4eajaWnnTahcWxhFu0fygkY5PpXz58aPg7Z/D34pW2gWV00lrd7HiaTG5Qxxg49K+h9W/ZD8IRfD+aRZLkarHamb7WJPlLBc42+lVKVHljdehVo2V2eN/ss/GrRPhlcaomsh44rrbh0TJGP6VkftcfGbSvipqemxaPGxtrJW/fOMFifarv7Lvwn0f4i+LL+LV/wB7bWakmAHHmHOOo7Va/bF+C/h/4brpd/oUbWiXZZHtdxZQR3Geac5Q59veBpX31PlOQDknrUXXg1JMGycioj0OK5HLUzLNqoVgRya+1vgT+1VoHgn4d2Gi6jazC4tVYbkHDc8V8S2bN5gB4Ga+7/2YfgN4Q8V/DuHWdXt1vrq5LDYXx5YHFbRcFC9RXQ9Las8Wt/jYsfx2fxt9k/ctPv8AKHXbjH8q9O+Of7VOmePPAN1oml2UyS3W0SSSDAUdeK4fWPhn4fsf2jF8LRSbdIa6VfvZ2qRkrmvdv2hvgv4J0P4VX1/Y2MFheWaBoZkbDOfQ+ua65TpXg3HXoD5brU8l/Z5/aWs/hj4XuNH1CwkugZTJE0R55HINcP46+NUvir4wWni5LIwx28kZSE8kqp6fWvZv2Rfhj4U8WeFb/UdXs4b66E3lhJjwq49K4f43eB/Cnh343aXpGmosGn3DR/aI0bIjycEA9qXNBVGox16lrlu9dTqvil+2FB4o8B3+kWGiz291eQmJ5ZGG0AjnHFfEt9EGYlvXvX6dfFT4W+AtP+Eepj+zLC2S3tC0FyqKJNwHB3dSa/MfVgWmcp03HFccpQlD3I2RCtbQy5EA6Co/LUfWpSpX72c1DsfcT2rjZIeWOhpfLHakKluBTlRk4NSBPaxgsMjvXt/7PHxM1L4X+KGvtPszfNNH5bQj+IV4hbxsWzngV9X/ALD82hL44u21YQCRbfMBuMbQ2ffvXXR0u7X8itLamT8dvjBr/wASNTsP7RsW06O1+aOAgjn15r0LR/2nvHK+CRaQ6MJxDB5S3kaMSABjPpmtb9sTWfDV9qGgQWT2z3iv++kgAOEz0JFe6+DvE3gfQ/h7ZxW15YCzhtQXjyuT8vOR3Nd7n7ibp38uxTcLI/ObS/EmqaV42j1+M51Fbj7R8wzls55r1v4tfH7xl8QPC4sdV077BZyjh1QqG/EjmuSsNU0W6+Nkd1MsSaK+p7iGGECbv5V9VftMePvBt58Lp7G0ubG/ncDyBblW8vHpjpW/O+ePuXffsZzaPmT4H/Ebx74Nj1Cz8I20l9DMA8sCQmTBHfjpXl/xX1zWvE3iS6vNdV1vs4dZAQR7YNfWP7HnxS8IeEfDeq2Or3UOn6g03mCSXA8xcdAa8H/ao8X6V42+Il5eaMsf2cgKXjxhj68VlWcpKS5bLv3Mrq54FNGoztA96rnavXGatyWnlsSWyaqTWpk/i4FeKyyNivpSR+WzjPTvQylVxUaW4TPOTUgT3HkrJ8h4PrTfxGKja23YJNP27eO1AhfMVhhTmjKpy1IkIjyQcmkeIOQTVIkdu79qb525cClK44xx60LCidOtMBk0gjbaSCfakLhselLJbqzBjSbBjFO4CLJu6dKcsyxuATTvLWFOOSaYtuHkDGrQHW+C9DvPFGvWWn6arPe3EipEq8fMTxXsfxo+BfjX4caRaXeu3v2q2kUDCy7insa8l+GfiqfwL4w0zW7dFlezlEmxu+K9q+Ov7Sdz8WtPhtXtDboozt7V7GHUuW6laPXuZu3VXIPhD+zj4x8feEbrXNLuxZWHIAaQr5uBzgV5zpPw31zWvHyeGbeMSaq9wYRhuhHU59BXrHwr/aq1LwD8OZPC/wBi85F3eVOoOQD615n4Z+LGpeG/iNF4ttUV7tJWcRtzuB6irtJxu5ehpddjpPjh+z34m+FNvZzazfR38EvQxyFth/GtX4d/soeJPG/w+n8URX8NhZ7GaKJicyAd+KyPjh8fdT+LjRLdw/ZlAB8utHwn+1R4h8NfDQeFYrbdbwgotwv909jRKMrpc+vVk+fKfPPijTpdD1a6sp1xNC5Ruc8isQ/MMkYre8QapJrGqXF5OP3kjlqxpmzzwK8WpbmaQyuJPmxg4ps0m3pUopjAdTWQxqtu68VGrEscdM1OcNzik3beBQBE7bcDGaXpzjnFOzhgaOpBNADEZ27EUpLKwAGalHzDApFbY3vSuAyRjt6c1LahsDK0MMtuNWbdTIwVBuPoKuO4H1P8J/2afDfiT4RzeKtb1No7pgfKhjfGK89+FPwftPiN8TH8PPf/AGayTc3mt1IBrnPDLeNv+EdkGnfbm0pM7lXJQDvVbwOviW48Up/wj/2htUOf9TncfWva5afKlZ3FHm6o7v4/fCfQfh5qVvbaTPvcnZJl9wJ9a9Duv2cfBmj/AAdtNaur3z9XuYRJlXxtJGcYrwLx/b+JbbWGh8QxTLe5ziX+lb994Z+IkfguG9uobxdE2fuzI3AHsKq9Lmehd5PVHTfs2fCHw38QfEWp/wBvXGLKxUlYi2N57Vwfx88K6L4Z8WTQaGdtryAmc4qL4c+H/GGuX1ynhhJmlC/vmRsDHua5rxrp+q6ZrE0OshxeKcNuNc1b2fsbpO5Kcm9djk/JdeT3pgjaRuOBVpmD5xUasN2K8GTNRhVtuB1qeKJlX5qQsF5NTRsGHPSsxixqep6VQ8TTBNEnIG04xV/eN2BxWV4sb/iSzdhxRHdEy2PHr9t1w1ZU2WlNaN4xNw/1rMbmVq7Wc0T6Y/ZttxD4PnlHWW5bP4AV7VbNnmvJPgDB5PgG1IHLyOx/E16zbsFr4XFvmrSZ9RQ0po0lOFprcimq2Vz1oZq4ToMLxvN5PhfUWB/5Yt/KvhPWWD6jcHrlzX238T7j7P4N1Bs9UxXw9fNuuJT/ALRr6PK4+62eVjOiKjGkP3qU5ppr3TzAzRSUUAeu2rV7D+z3ci38eWQLbQxrxy3bgV6X8HLo2/jTTXzj94K8vEq9No9Wn8SP0SsWLW6n2qX+Kquktus4z6irP8VfCJW0PYJKawpfamtVAeb/ABziLeB751GSq7vyNeEftBr9u+A3h+8x80N2i5Hbcp/wr6K+LFt9s8H6hH3MTY/KvBPiREt9+zPJ5il/Ikhfg9CDtz+tepgZctWL8znxCvTfofIsmXUe1V+Nxz6VYkYDpyKqt/rMZr9IifJMs2m3zVLcDIr2zRF83S7YjpsFeHwtsZTnvXtnhX5dGtlznA6+tZV9jSjuzS+xgMCCan8kSLj9aEn/AHm0jipWbaMjrXEdZDHD5ZwKc1uHzk80RSNI2CMGiZ2TlRTEhyx8AHg00whWzSoxbk03zmDYI4oGDRiTOelNWEL0FOk3bflFNjZ2HzDBpki+SjLyPmpSo3YxxTXWTdlRxTmyy+9MBjRquSOKTyxIMsKTEu/5hgUsocrhRzVAIIwq4AxSeSnpz60Lv2/N1prLKxGDhc00BIIwxwx4q3bxBTxVbacYHWp7VGRhvrWO40fVPwd/ao1fwD4Ri0htJfUYIBhGHG0V5Pe/EjU5vijN4vhXyL17nzwo7c9MV9a/sh2fhC6+GqLLBYzam7EXH2hVZiO3XtXgXxO0/wAL6R+0dJbW6xDQFu4zLHH9wZxuUe2c16lOScmuWzHHlu0b3xM/as1/xz4FuNBm0tLQXSqslxGTkjv271yfwN+N2v8AwtNxDp0H223m5a3cEjPrX1x8XdL8C6h8GNVeGDTViis99q8YQOjgfLg9c14b+xfdeGIdY1N9XNuNRwPszTgYA74z3pRmnFvk07C93lueOfF34maz8SvFEWqarH9muIkCJEBgLg5AFei6D+09470fwTJp32P7XbLEYlumQkqpGOTW/wDtqXHhv/hJ9Ck0z7K16yFrt7cg5XPGcd+te8fDDxD4AuPhdaW0cunpb/ZMXEEm3JO35s560nP3E+T5BeNlc/M7VWa4upppTmSRyzfUnNZhxniun8crbN4m1X7DgWn2h/K29Nu44/SuZ8sq3tXHU3EKmN3PNdZ4D1+/8L+ILTUdMbZdwtlMVyJUs3HFdt8LdQs9L8aaTcaioe0juFaVT0xmqov3tB6dT2b4nfFT4keLPCK2utwTW+lS4LOISqv6c1j/AAX8W+P/AA/59p4QE0wf53hjj3jPrX1D8ZvjF4G1L4Q6naQXltdyzW4jgtlxuV+MEDtivOP2PfiV4c8J6bqtrq1zFZ3U0iss0ndcdPauvmqcrah8gvHl2PB/iJrfinVvGTXviRpk1dCOJE2FMdAB2Fen3fxA+MGpeAHRYrx9DMOxrlLckbMc/N1/Gl/aq+IXh/xd8QtMl0iRLqK1iVLi4QfK/wA2evfivoex/aE8A2fw7Qfb4UMdn5f2Hb8xO3G3FLmqJK0QfLZaHxj8J9Q8Xaf4hx4TM76lMMFYQSSM96k+PUnj+41C2fxvHdJKFPk+cMLj27V3n7M/xW0DwP461a81bFta3asI5dudnzE4rd/bE+Nfhjx54f0rS9FlF7LHMZWnA4UYxgVpUlU25dO4nJc2x8eyzBWIqBh82aluIQXJJ+lR7e1ea9ySa3ZWOAea+gfgjovxO1bS5F8JS3IsY8ghWwg9ga+fLdNr7q+1/wBmX9ozwp4D+H0Oi6sXt7qN2csiZ3ZropSmk/Zq7K2V7Hz1r+n+ItM8bSx6iLiHX1l+YtkSbyeDXofxF8CfFCDwXHqGvG8l0gBSyySbtuehYdqo/ET4tWHij45ReKIYMWEM8WFYcuqdzXtHxl/ao8NeJPhvfaTpUckt3eRCIh1wqV3OVayUV6ibV1oeKfBXwH478Wfak8LTz20Cf62RZfLQH0zXNfEbwR4m8K+NDZa+JjqrFWV2ffvz0KnvXsP7MX7QGhfDXw9f6frEUweSbzFeFN2eMVyHx0+NVr4/+Jmm6zYW7R2djsVBIPmbDZyapSrSlZr3S9L2saHi/wCB3xR/4V++p38802mxwiV7Vrks4THUr/Svl7Uf3ZKkcg4NfdXjj9srRtR8A3mnWmmT/bbi2MGXxsBIwa+GNQ/fSO5HVs4rirqrb96Z8ytorGSzbuTUfmAsRU8iDovSoVjXPSvPYhmdvWnpIGoKjce4pdg3DA4pAWLeQbgK9y/Z1+Fd78UvEzWNrdGyhij8ya4XOVH4V4haoncV7h+zn8Xbv4Ta/PdwWn21LhPKeId/euugpN+49S4+h3fx++Adz8MZNPuP7ROo212xRXYYYMO1eheDv2O7nWfCEF5f601leXEXmJAqFgoIyMnNeW/HP476t8TrqwjmsfsEFo/mJHzya9E0f9rrxHb+E47WPQ1eSGHyxdYbjAxn0r0bYjlspa9Ryk7LTU+fW8AXMnxE/wCEXWRGujefZhIMgZ3YzXtvxc/ZKk8B+BJdah1v7a9uoM8Lx7B77Tk14La+MtSs/HSeIshr+O6+04boWznFet/Fj9pbxH8QPC/9m3emrYW8wG4oCA351qlU5lyysupnJ67Fn9nX9my2+LGi3urX1+9paRSeSixLlmbHJPoK8f8Aj98NR8KfGVzo63X2uJfmSToce/vXZfBb4ueNfAMF9ZeHbd76Cf53gERk2nH3uBxXmPxS8V6r4w8TXF/rIYXb8MrDBH51jVUrScpadEJt9tDzmSYyN0IqpcTFCMDOavXGOccVXdV4JIzXiyEVWfauSOaihkMmeMVabBz3prMm3jGfapAgkkZWxjNKznaD3p+4LyetOUDq3IpgRRSM2SRxTjuJ+Ucd6eGU8DBHpT+Fxzj2pkEQYr/+qmQ72ySvFWNoJz2oZlbhelVYCnMz7gFBIpWzs96nMiq2DQ+ByelFgK8e5lO6nKrNINvSpo8Se9SxxhG96tID0r9n7RNG174naLY+IZVj0uST95uOAccgE19C/tfaH4D0vTbSPw9a2sFzDgbrYjDCvmD4e+C9V8c+JrPStHQtfzPiMg4x713vxm+C/iv4X/Zh4gnW4WUZVlkLAe1exQVPrq+xFpvbY+g/gHofwyt/gfJf6nFZ3uqzo/nedguh7AeleH/B2x8ISfHqGHWVjXQRK5RZvuEj7oPtSeDf2d/GviD4eHxNpsvlafJnbFvIZwOp+lef+Dfh7rnjTxtF4e0+M/2k0hBJONuDyTTfs7O19d/I197qe6ftjXXgy4vrQ+H7Szikhwu61UAH8q7jwhc/DHQv2ebcSQWF1q93ATKXUNJvr5w+M3wf1/4W6pFa6vcx3m4YEkbZAPpXYWP7KPia6+FyeKn1GO1gkj8yO1ZiCQeneqvSuk07fmDcl1PnfxRHBJq919kG23LnaF6AZrDZCqkda09Ria1mkicYdGKsKzm6V4tR3k7KxPqQxwFmLZ/CkkhLEDOBT42O4/KcUSM24ALkk1kA3ywqkd6ZHDjJ3c+lTMrKM4+tRrubOVwKQxrQh2BJwKV04x1odWDYAzTsMq5wc0gsJHCFXIJoEY3ZOaURybeQRSqr7sBSRVDFELTZCKSB1qW3TY2VOKdHcT2qssRxuGDxT7aJpGVW+VmOM1cVqhM+qfD37RHh/S/g/a+HrewWG6WMrK5Qcn61wvwJ+MFj8NfG9/rFzaeaJ0bY2Pu5Ndsv7O/hvS/hLBrV3qZfUriLzAFcALx0xWJ+zL8JfDfjvxBqreIrkGzsUO2LdgOfWvf55xUVy/8ABIVNbcxyPxe+KUHjvxVDqaRhkSXft9gc4r0P4jftPWninwJaaLZWvlJDCIyuMYIGK4P4seDfD2l/EKGx0dvLsZJghXdkKM4ruvjB8I/BnhDwRbS6fOG1Axh2kDdTjpT5qsnKSitBpRjpc4r4J/HFfhfZaxC1tua8HDgdPavOPiJ4ubxhrkt4yldxyK+hP2d/hv4H1fwDqGueIGjnvhuSOJ2GBXzv8TbOxs/FN5HpvFsHO0A5A9q5MTKr9Xs1oKMYRldPU5VlXtwadCibjkAmodrDqakjUsCd3TtXzkjcc8a7sYoYj1qJmZjjNDBh05NQBJkdP1rH8Zbzo4VeMtyTWvDGw5PPNYPj5gulxqc8vxinD4kTPRM8luv9e/1rOP8ArDn1q9M37xz71R/jPNdsjnifWnwUj8r4f6X/ALSbvzNekQdq4H4TwfZ/AejpnP7hTXewH0r4PEfxJep9VS+BGgpAX3pGemqx20jVxmpwfxquPJ8EXgB5bAr4wumzI31r64/aAujB4OxnG98fpXyHJyxya+oy2NqR5GMfvIjzTTil+6KbmvYPPEP1opfwooA9Zt67j4e3Bt/EmnyK2Csqn9a4e36Cup8JyeTq1o3T94v8xXn1kpRdz0oPU/Sjw5MZdLgbOcoD+laX8VYHgmYS6FaMDkGJT+lb1fBtWbR7fQfTWPFBpppCZz3jqH7R4dvExnMZ/lXzvqYN5+zX4ij2ndCoIXHpKtfSfiSMS6Pcj/YP8q+dNIU3HwZ8eWpbBihuCD/ugt/Su7Dvla9UZ1NYv0PjWTK7vTNQN97pg1duFHmPk559KqMvzV+mx2R8dLccmAQfevZ/AxDaHFgHAJzn614uCGwOleu/DiR5NDx1Ktz+VTW+E0pfEdgqruHHNPxtbpUBhdmDZ/Cp9pZfeuA6gwGye9KKijiMbHnIp0kJkXIOKAHMnSgqpPOM0sakLyc01rYlgQ1NAOUBaadpYnrUwtTIuKeunmPJHStFHQorMwHB4poUdqt/2aZmI5/CnrY7AR3p2aHyvsUmkBYDuKTOxcmrTaZtbdg5NP8A7PMyhe3tTsw5WZ+Q3vSsy4wetXf7N8nAwc+9NOm7ssetUosfKymoqzCwkkAp62ZbCgdanWxMLKSCD2raMXcFFnt/wM+CvjH4iWNzdaHdnT7WM7TM7lVZvQYrjfHnw+1zwh42n0XV4mOp7xg7t3mZ6EHuDXq37Of7SifCnRZdI1Gye7s2cyKY+qmuP+NHxjHxE+KUPiOxtntobZY1ijk+98pzk13xVW+uw09djstb/Zd8e2PgGTVpLvzLeODz5bATneEAz0xg8dq4H4L/AAr1r4m+IJLLR5VtWhXfNM7EKgzjt3r3W/8A20or7wPd2B0eRtQktWg8wN8oJUjOK8Q+BfxjvvhP4huL6GD7THcrtlhI+9zWkfbWfNv0J5nZ6F747fA/XvhXcWUuq3aajb3mViukJPI6qc9+a7P4d/soeJfGXg2HVo9Uj0+O4TfDA2SWHbPpXM/tEftAXPxhXSrdLD+z7ayLybSclmbH9BXXfDH9sS78EeELbSLzTTffZV2ROD0AHANQ/bctk/eFeSWx8x+MNEufDOvX2mXY23NrK0Ug64YHBrn/ADA2cV0/j7xBN4s8VaprM67ZLydpiPTJziuZytefO9yHuNZ9rA4rofCGky+INbsrCAZmuJFjTPqTisBQNwzW94X1KfQ9Ytb21/11vIsi/UHIop35kXHc+s/Fn7F8+i+AbnVYNW87ULWA3Eluy/IQBlgD64zXAfs7/A4fFvVb1Z7w2dnaAGV05bJzgAfhXQ+KP2rPGGveCrjSjYLAtxF5Ul0oI+UjB/SvP/gz8UvEXw51S4fQ4BdSXGN8O3Oa7v3mqctS/ftsb/7QHwPT4Q+INPt4L43tlfIXiaQYcEHBB/OvY/DH7GOk6x4Ht7u81WePVbi3EieUAY0JGQDkZNfPnxm+JXib4ja9bT6/B9kltV2RwBCoXJz0Nd7o37SPxGtvB/2CytxLawweWLoQlmRcdd1HLUlFWlr1C87Ky1OQ+EfwfPj74my+Gp7jyoLUv50qeinBxXZftPfs16V8LPCtrrekXs0iNL5MkVwQTkjIIIHtXkngHxt4i8K+LV1TR5HfUpWKngsXJ6/Wuk+PHxA8f+LLKzt/FdvcWdop8yOOSExq59eetVUUuktOxMube2h4JcTMGwBmj+HPSpJWVc+tR5GPQV5zMh0DEsAa+xv2Vf2eNA+IXhOfW9bLzr5pijhRsYwBkmvj2Bl9ckV738C9f+JEdnLp/g5Z5YVy8ixjIBPrXRTV00pWZcb/AGS/8WPhLpfhH40W/hmxnP2G4eHDSEZjDnBB+le+/FT9mfwVonwu1C9tI2t76ztvNS5Mn32A6Ee9fJnjRPFP/CcTP4gFxHrwlVsSA7geq4/SvT/G2j/F648BtPq6Xz6KIw7gnouOpHXFdHIrL3/+CX77sbn7Jvwb8PfEKPVL/W0N1HausaWu7AJIzk45rA/aO+Fvh/wJ8StNsdIk8m1vlRnhZt3lEtjr6d64/wCDMnjuS+ns/Br3PnyDfIlucDjuayPiZpni6y8XbPE4uotVG1gbk5OOxB9PpWkYqM7uXyB81z7X1z9n/wAAWnwtuIX023RorMyDUGbEm/bkNnPr2r809bj23kyRHKhiBivqTUPhb8XtQ+Hbai01y+ltB5ptftHz+Xj+59K+W9QHkyOH+9nmuapGMYu0+YytJLUx2Vk+9xUGHZ89qtSsJORzVdpFHy964GSMfceB1p8asvWkbEa7qdHMGxUgWLdTuz2r6Y/Yrt9DvviPIutJBIsVuXhW4xt3568+1fMK3QV9vetKw1u60mfz7WV4XxjdGcHFdFNq9pbBdpaH3D+2T/wiqx6ENOFmmotKVkNsFHycdcV7h4IuvA3hn4b2kEc+nfY47UNMCUYuduWz6mvy5uvFF5qzB7qeSZ16FySRUi+Lr6OFoRcSeWeNu44/Kuz9zKKg5OyE5yaSPTtK1HRLr40QyTLGNFfVASrDCCPf0+mK+tf2ofGXgyb4WzWMMlleXAC/ZhAVPl+mK/Oz7c7MW3HPWpP7buJoxG80jxj+EscflT9pTlJSl0Fdn2l+x38VfB/hXwxq9pq9xBZai0/mCSQZMke0YAPtg/nXgP7UXijSvFvxIv73SEQW7nhk4BPrXkM2ptGw2kgn0qKe4Z/mJycdzWU6tP3mlqxalQxlWJJqvNCZmB6CnrcGQng8VFLMVxtGSa4bjFaP5dpqOO18vk1KWKqSRUccxbIINIBrW/mMCSQKmW2L8KPpTVkZWAC5HWvRfg34RtfG/wAQNE0W7mFvBe3CxPJ6KcmtIx5tifU4GLTfK5PXuKeNOMjbypOK+0P2ov2ffBfgHwrbXGgB4b6P/WK0m7zPw9as/ss/APwR41+Hd9rfiIfabjznh8vzdvkgD9Sfeul0XGCnbRlLlZ8Utp7bSAPamDTfLXIBr6A0P4Z+Hrr4+2nhqS5K6JJdldznB2ckLn3Ir1T9q74S+A/B/h21fw/bR2l/GQGET7gwyBz71p9XlzcltQ9217nxYuk+Z821jjrgUHTy2UC9eMV97/s3/Cr4e6h8HG1nWLa2vb+bzBKZiMxYyAAK8I8D+DvC+pfH610i+dY9Be4ckMeDjkKTSjQlJN20Qvd7ngo0z7Ooyu0VOum7vnCk/hX1t+1t4N8DaHa2I8N20FtOnD/Z+jc967X4O+D/AIa2/wACIr7Ube0vdUuYWMxlwXV+cKPTpWv1efu+69TNyit2fJHwn8d3vwv8aWWvWUSySwkgow6qeDXafHH46XXxauIzNAYdvRG6CrPwf0nwlP8AHCK21zYNE8xykch+Un+EGuo/ath8HjVoD4bs7e124Q/ZxgHHevRpQqQUoqOqW5EpRursyPCf7UGr+F/hnH4V+yq0cSFUmjP8J6Z968x8EfEzVPBHjseI9N2/ayWyjZOQa+q/Bb/C+1/Z9t0NtZ3mqz22bhnAMok789sV4l+zPJ4Nt/i5cSeJFibT4UdrZZgCm8N/hU88+VWhb9Tb3bbnF/Fn4qaj8TNRWe+jMbjnr3rrdQ/aO8XXXw7tfDzW22xt4RGJlBwcdzU37UWueHNa8apc6LBBbwdG+zqBkZ9q9x8Y/ED4daf8DbHRtMisZZWs1LBAu7cRzz1zmtakq3OrLX8jNU6bXvbHwFfzNNM8kpy7Ek1RbDcir+qxrJeShDhN3FUWUqNuc185K/M7l6dARQeB0rV0nw3e6w5+yW0tyUGT5aFsVm28HQg19Xfsn/Fbwh8P/Ceu2+s28T6pJJmMyKD8pHatqNN1XYpW6ny7NpUsdx5Dxss27b5ZHOfpWjq3gvVdDto5b2wnt4pOVaRCAa9OuPGug3Xxs0/Xbm3T+yo7tXmjA4K5P+I/KvT/ANpz47eG/iBpMNhosEflKoVNqgYxXYsHdtX0QnKJ8w6P4M1bXIZZbDTbi7jj+88MZYD8qr6f4bvNU1JLC2tpJrx22LCq5bd6Yr6i+DPx68N+Bfg7Poj2qLq8jP5spQZb05rzf4W/FLTfCXxqt/FF5bCSx3OSNucE9Dj8P1olhEknfcXOuiPN/Ffw913wW8UesadNYvIMqsqEZqzpfwv8S6poL6vaaRczaegJadU+Xj+det/tLfG61+LN2ptE2oDlSR0+lb2g/tKabpfwdtPDMUZW5jj2OCMEt6+9aRwSvZyBTTd7Hzv4a8D6v4v1lNM0myku7xv+WaLz9T6VL4n8Eax4K1L7FrFm9rcA5CnmvSvgP8Yovhf441DV7i23rcxMiYXJQ5JrI+L3xSX4geJlv40OxH3dMVmqEOXm5htq2xWu9D8dx+D4r64guhowX5Wc8bfpSfCzwV4y8VXV2PCqyqVX9/IGIXHvXofjr9pWLxH4HtdEt7IQCGERbce2KxPgf8fn+Fuj6xZJb73vTuDj6YxXTKPvx9/UlNLZHm3iDw7r1v4qGm3yyPqfmBAuckt2xXVfEL4c+N/C+k21x4g3i3cDarPkgY7isrUPiVNcfEC38RmPfJBMJFVu/NdZ8Yvj9dfEq1jgMBhTGDu61nZJSfOUpJ9Ch8O/gv4x8VeF7zVdKkNtpaZzucgP9BXkHiO1udL1K4t7skzoxBzXt/gv9oXUPCnw7bw4kZKAtlwDjmvEfEWpNrOpzXUg5ck1zYrl9jG0rsmN29jI87IpI7oHjpSyqG6Coo8LkYwa8NmqJ3m7AZp+4qucVHb43gn9ausiFSWIB61mMghZpFBxiua+Isg+y2idzuaurhG5eB8ua4/4mLsW2P8ACEOM9aun8RnU+E8sm/i9aok7UY9ODV2T7rH2qvHH50ip/eYL+ZxXTLa5nBan2V4FtfsfhnTYR/BAg/QV1lu3Suf0NfJsYEHAVFA/Kt2Bs4r4Ks7ybPqaeiRor9wUjUisCo5pG5xXOUeM/tIXSw6DbRk/eJ4r5Zkb5jivpH9pybFvZRZ/hzj8a+bZOtfWYCNqKPHxT98i3etJjNFFemcQde1FJgUUAer27V0OhsY7qJx2YGudt+K29Lcq6emcVyVI3R6ET9GvhbcfavCOmsephX+Vdg3rXnHwPuvtHgfTTnJEeK9H96+BqK05LzPbi9EKaQ0AikNSBT1ZfMsZl9Qa+f8A4fWf2618f6TJEGSU3Mfln+JXTGP1NfQl4u61kGe1eGfD9lt/iF4pticBmBwPda3pbsTV0fDHlvJygZkC5PHt1qo3p+Faeq2/9n6peWrjY0UrxlfTDEY/Ss2THzZ9a/UaXwJnx9TdoY3y9K9Z+FsnmaPOpPKuP1B/wryUncRXqPwhU3EN7GDjaFbH506nwMVP4j0FXVJBuq3G6DORuBFVWUHgjJ9afH25rzjsGMdshAFK8hVemRT5FGcjrSKvtmmAxG8wZ6U/zNrAYJowBU8ZVlxVID0b4H/DlPif40s9ImuBawyHLyYyR7AV7b8f/wBlXTvhv4MfxDo+oSzxQSIk8E6jOGO0MpHvjivnbwL4i1DwtrVvf6Y7JdRn5dvXtXq3xO+MHjrxh4Zj0vX4pI9PkZXDNEU3Y6c4FejCHMk0ylzX0O5/Z1/Zn0D4i+EZdW1m7m8xpCiw27KNuO5JBry/4q/B+H4f/FgeGYrrzraVo3imcAEI57+45qb4S/EDxn4dc23hhbm4UHcYoUZx+Qrn/iP4o8QeJPGk2p68skOqjapSRChQLnAwfr+taexV730NE53PqLxH+x14Y/4V/PPYXkw1OK1M63DMDG5C7sEehxXiH7MPwp0j4ieMpYNYfNrbIX8kHBkPpmrcfxK+KE3geS2jW/OivCYzMsTEbMY64rzr4eav4h0bxFGfD5lN8x2qsQJYmhUUrpsOafXc9s/ay+Bfh/4d6XpWsaGhto55zby27NnJwWDD8sfjXY/An9mfwd4q+G1lqeqK15e3kZZmR8eXnoPrXiHxs1n4gaxFYJ4wgubeKMloBKhCk49fWtL4TXXxRk0OWPwml3LZR8Hyx8v0oVH3bc3zF7/LZGDH8LtL0/4/SeELi6DaZBfmFps4JTgj8cGvov4/fs+eB9D+FGratptqthfWEHmwyq5PmEEfKR718i6wuup4xuJLz7RHrrXGXDAh/MJ6Y9c16j480D4rf8ICza3FfNoe1WfdyAvYn2rTlTafNa34g1LQ6j9j/wCFvhbxpDql9riR3s8TBIrSRsDBzlvfpXLftUfDXw/8PfiNpdvoo+z2l/CJJbcPuEJ3YOM9ARzzXDfCXSfGGp6w9v4QW5a725YwkgAe5qL4vaL4w0XxFEPGC3P9oPGDG8/OVB6g+lU0ufmUvkHvc259raD8Dfhzqnwtt4v7NtZlez3vfg/vVfbktuzwQa+YP2Z/DHhvXPilLaa+Y5rW3L+SshASV1bAz6jjNS+Dfhz8WPEHgH7doz3LaNJGxSMT7WkUdcL3ry7wd4d1vVvE9vpujxzDVWk2KqZDq3Q59MUoKOq5t/wJXNrqfTf7aXgHwtoPhXStW0uztbDUWuvJK2wCiRNpOSo9CBz71p/st/DPwJq3gyx1y5Fvd6ttdLmO4YFeT02npxXh/wAaPhR4+8H6NZ33idzdWO7arrOZFRiOh9KX4N/Bvxz480ee78P3JsrJWIMkkpRWP90etK0eTl5/mHvcu5538eNJ0zRfir4ksdF2jTY7oiFUOQowCQPbOa82WNg3Jzk12fxG8L6r4L8VX+ma1G0eoQt+83HO7PIbPcGuQ84biMVx1LXsjNjWQ7uK3PC2xdYsluOYWmVX/wB3cAf0zWL5g3VoabmadEX7xOBjrUw3HE/S3WNS+G0PwqvIj/Zi2S2DKE2rvDbOPfOa+ef2N9e8NaT4q1efW5beCdYwLaS4AAX5jkjPtVfTf2SfF2rfD+PVzexpI8HnpZSOdzLjI9s15x8H/hLqPxP8XHRrOVbUxhmmmc8IAcHgdea7FGnZ2enUuzs9T2b9tXxR4Z1u58Pf2TNb3OoxiTzZIcY8sj5QxHuP1r2T4R/EvwDpfwp0uGS8sbMRWoFzAy/MzY+YkY5zXy18fP2e7z4Ox6ZdPqa6na3jGLeQQyOBnGCemP5V3nwp/ZCfx14Hg1i/1dtOa6QtbxRxh+OgLZ96T9ioWu7CtpucX8H/ABt4a0P4+XOr3MccOiPcTG3Vh8sSsTtOPbNeqftjfFTwj4o+Glvpul3kOoajJcpLGYxzGoPzH8RXg3h/4O3GofGFvBE9zHFPFdG3e4XkYHcfh/OvTv2hP2U7H4c+A5fEGl6nPdfZiqzRXAHIJxkEfWrn7JtN7ie61PkK5h3MeajZOwNT3UhjbpmoFk4ziuJ7kEtpBtbOea+yP2RfjR4Z8AeE9QsdZk+zXUs/m+ZjqNoGK+NbWZmk27cCvqL9lX4D6f8AFeHUb3VbiSOztdqeXH95mOT+Va0+Wz51oV03sVvjd8WtJ8ZfGnTdcsYy1hZeQrNj/WbXyc/hXvHxE/aq8KXngHUrSxDXN5dWrQLEV4BZSM/hmvAfjt8G7D4c/EjTtG065MtrfKrIJOsZLbcH+dfQXiL9lPwdYfDW6dWkXUoLNpvtvmcFwuenTGa6G6Notr0NHbS7PHf2WvjVofwzj1ePVoWDXjIVkQZKgA8frWD+0l8ZLD4keONOvdLgIt7CIJvfGX+bdz+daH7Lfwf0r4leItROsOzWdlGD5CHHmHOBz6UftRfCPQvhv4s0pNGZooL6P5oGbcVYEDr6HNac1P2llH3gaXNvqetX37ZPh7/hA3ggsrl9TNn5W1lATdtwa+C9Wl+2XckpGNzFuPfmv0W/4Zm+H3/CsFaa1U3RsvNbURKQ27ZnPXGPbFfnRrkAgv54Ym3LG5UH156/pXHUlCz5I2MtLaMy5FCnC8VAY16nr61IysPvAg1AyybiR9yuUkVgGODyKXywoyoqOTLcKfakUPGPmOakVyVYxzkc0itzg9qi2uTnPHpSlWbIGR2qhFhX546d6esoZdpxVVIjDnJzzTfKdmypp3An8z5j1pzMFFV+dxx97FNjhZCWZs0AT8dTjNNY7uM5qKSAyyA7ttKynO3OfepAQkKMD8aZuVcZ602O3MbcnrSyWvmMCTgUABb5valGzPByaPJypGfamRwLH3yaAJ42VW54PSuo8HWeoX2t2UOlLI1+0iiEQ537s8Yx3rl1hErhm44xXovwb8bD4d+P9G154Bcx2c4doz3HQ110L82m5DPSPi14V+Jvh7RbSfxXBdLYuAokkYNg8cHHTimfCfwN8SfEXh2+uvCkVw2m7v3pSQIjMByOTyRgdK7/APaM/aY0z4meGU0zTEIiPLKeTnj/AOtTfgV+1DbfDj4bt4ensT9pjlkeObPHzH09c17f+0OzbXOKHK1toeC2ug+INQ8bJp0cc7a81xsWMcSeZmu2+Mnw7+Ifg+ytZfFheSGbAVxLvA9jWXofxam0P4wweM44FmMVyZTG38QOQR+R/Suz/aC/aLb4t6fDaQ27W0KMDs9+tCjO7SkuXqXKy6GT8LfhH8RPFXhO51LQvMi0f5vvSbA+Ou0Vw/hrwTr/AIq8cR6JYQSNrTTMCucFSOpJ9K9f+GP7VVx4H+F6+F/snzwK6RzDrtY//XrzbwH8XrrwP8Uh4tghEzbmDo38SsMGlapy6teX/BE7djR+N3ws8YfDlrMeI7j7THOMqyPuA9q2fAf7P/jrxN8PZNfsbgWmmFGeOJ3I8wDuBWV8d/jxe/FieEtEYY4yNqk8VseGP2oNd0P4ZxeF1t2aG3jMUcqjopycfrWlqnMvfV+rISTWsTzvwB8Odc8eeN4tD01D/agcs0jHGzB5Ymtb41fCvXvhfq0FprN3Hd+Zz5ivkZx3qh8Nvirqfw/8cf8ACQWQ3XL7ldMZyrdRR8Yfitf/ABK1JJ7yJ0dW3Ykzxmoez9/3fzZWl9j0Gz/ZW8UQ/C1fFbanDbRTRefHYsxDbSM5J9xXB/A34J6n8ZPFk+lWNzHZi1TzJ7iQEiMZxwB1Oa3br9ozxRffD210K4WR7W3jECyDOCoGBn8OK4v4Y/E7XPhvr9xe6O5Ml3H5csa5G4Z4qOW0UnPX8hXd9i38XfhXe/Dfxgujz36akZDtSZARuOcYI7V6b48/Y8vvBvw0h8QXGvRvePGJGtdpCjIzgHNeLeOfHWp+LNaF5d5SZGBGRyD/AJxXXeNvjt408UeF4LTVFmW0WNYllZSFYDjINOSg5u09OvmCuuh4XfF4bhkI5U4NQl+MkVZmIVs5zn1qu2GbOK8GW4xbaRjk7eO1fTXwA/Zvsfid4H1HxDf6j9jihdkQL14H9a+aIvY16z8PPEnjqx8J3Vh4eiuZtL3l5RGpKqcc8124Xl5vedio3voh3h34Uxa98YrHwiL1RbT3JQ3BwPlGT+eBXon7TPwH8NfCy3tjod9JcOABKJGBBPfFeJaXfa03jC1ubTzv7ZSX90qA793IxW/8Trjxi00f/CTpcRNJ8yLMCK7bU3dpsnme1j1j4c/s4eH9a+D6+LNW1Yrd3EbPFBGRlcetcV8BfhLo/wARviu+g6peCHTLdXd2JwZFB4A596w9CtfHz+C3k0+K8k0JMncqnYPXFc54EtfE954kT/hHBcSau4YDyevPWiSp2SKbl0R6V+038NvC3w/1qODw1K/kj5WRn3/jXa3nwE8DaT8HrLV5b2SbV7qASGQSkBc9OM4/SvAviJpvifS9YW28SrMl2OR5ldBfeC/iFb+CbfUbqG7GheXmMscfL6+uK0UqPM7p2X4kctQ7j9lX4b+EvGniLW5fFDq9rYRbooWOA+T1rivjF4Y8P2PxCSz0XbBZSTiMqhyq5YDP6/pVT4T/AA18ZePru9PhUSIbdMzyq5VcdcE9+1c94n8L65pnij+zNRVn1PeI1CtnLE9q5r0+W6WpquZo9s+L/wAKfA3hPwbbNpkvmaiIgzybskuQM/zq3+zj4L8DXXgHV9Y8RJFc6gkrRRRSH7o5xx715f4++GPjXwfodvda6G+zOqlQz5IGO4/KrPw4+CvjDxl4XvNZ0yY2emRsQzM5CuQK3cqTmrRM5Kd/Mi0rQ/Dt58ZrWzby49GkucFWOVHFdJ+0do/hDT5IP7ARFdCAWjPUd815l4e8E63r/jRNEs13amZSu7d0IPXPpWn8U/hprfw/vobfV5xcM/cNnB9KiUocklyahyS+JPQ918H2nw60n4F2sssFrdaxcxZmkkAJ3Y5FfIWvLC2qXX2fiDzCFx6Zr22L9nfxKvw9j1+e9W2t5E8yO1Y9Qa8KvFMMzIR8ynBrz8XbkilGxevVlJkKrxTI7faNxbJ9Kst92khVm7YryGMasJbBzipfKLcHmnhjuwOlSN8qDA5qBjoIRGoyfrXBfFCbFxHEOSsOT+Nd4sh288GvPviRIk13NubbJHGoUevT/wCvV0viM6mx5zMy/ZwAmGHVsnnJqXw7bi68QaZEej3USn8XFV5/9Xj3rV8Crv8AGGjg9BcoT+BzWtT4GTT+JH15YsBCg9BWxbtXPW99bxxgvPEgHq4FWP8AhJtKtceZqNun/bQV8POMpPRH0qkkjp1bjihmPrxXIXPxO8N2akNqcbEddvzfyrIuvjl4ZtzxNPJ/uQtUKhUe0Q9pBdTzn9py5LaraRg42xjj8zXgUgPJr1P4weLIfG2rG7sQ4tYwq7pBjtXmf2VmyAVP419ZhYuFJJnjVpc020VMetG0k1a/s+b+6TUclu8fVSK67nOQbaKXbiigD1K3rYsW5TnvWHb1sWZw3Fc8jvifeX7OsxfwPZrnOM169mvB/wBl++87wmEz/q3xivdt1fD4pctaSPZg7wQ4UGmg0bjXMUMueYm+leGaCotfi5q0W3BliDf0r3Kb7p+leFSXDWnxtWPgCSDHT0Jrenu7dgb0PjX4iWa2vjzXkBGFvpgBj/bNcxIuWrufjPZnTvip4mh5x9sZwD/tAN/MmuHkO481+k4V81KL8j5GtpNkRyCPevSvg/dNDe3hHP7scfjXmx6DNd18LJBHq0gYtyhPFdctYMyh8SPWLjewDrxk9KWP7oz1pnnIF68VIjBuRyK8s7hFjfduzxUrKSCF61GLhVbaTTvNEak0wGorR8HmpBGzn5eKYkyy1Mswi47VaA9E+Cmq6fo3jzS7nVYVmtI5QzK/3a+0/j9468F618HtURbi0upniH2aJQpdJOxA7Yr4M8J6PN4i1a0sbbPnzuEQ9OSa+mfFn7HusaP4Kl1aLW47u7t4fOktWU4IAyQGz1/Cu6PJZcxVtrsk/ZH+I2geEUv7XUzHbyTEYnI5/wD1VhftZeKvD/iL4gaVcaQ8Nx5dvtuJosfMd2Rn6DNc/wDAX4Kv8XNSuVfUf7PtYFyzqNzN7AZp3x6+BM3wj1KwIv8A+0LG9yscjja6sOxH9a6YuHPfqVb3t9T6Y8D/ABz8Ft8O4ILuSK2a3tfLe3KjDYXHT3r5d+C/j7SvBvxYl1e4gX7E00pjGPuKWOMfhXq3gH9kK38UeB7bUbjWpILq6j3pHGgKL6A14n4b+FM2rfFSTwhcXS28kNy0Mk3bAPUfWhey5ny/MSS1sz3P9qv4yeGPGnw7t9P0m4W6u3uUkyF5QDr/ADqD9mv9oLRPB/hVdF1f9wkbllmUevrWP8cf2WLT4e+CZte07VJLg27L5sU46gnGRVT9m/8AZ90j4oaLeajqt5JGEfYsUJw31NJOlyNfZFZcu5xfxS+JGnax8dpPEtggayimiIyOH29Sf89q+hfGn7UfhTWfhrqlsuXvrmzaEW+3jLLj+tfOvxo+DsXw5+JFtoUF551rdhHikfgqCcYavoC9/ZD8MN8PXuIb2YaoLXzhchgYy23OMY6USlTsuZeg/d01PEv2aPjRY/CfVr0ajCXtbpQGZByPSpf2rfjJo/xU1PRRoySGOzV98jLgndjj9Kofs4/C7SfiN46m0/WZittboxMStgyEHGAa6T9qz4E6F8L4NK1LQ5JI4LqQxSW0r7ipxncPaqcoKadveB25ty18N/2rG8B+FrHTorNLq3hTY9uc5Bx1U54/WvMvh78Wj4W+LU3it7dZEmuJJmjx2ckn+dfRfwJ/Zv8ABPir4a22oaqhv768QkvHKV8n0wB3+tfOlh8PdJj+ODeFZbrGlw35gMzEDKA+3erdSE5O0deoopapM9Y/aG/aW0r4kfD86Hp1pJHLNKju0nRdpB4rI+AP7TifDDwy2jX1i11ArmRDGMEZr0f9pT4G+C/DPwrudV0e0jsL202FJEkJ80Z5B55rn/2RPhn4P8WaFqF/rUMF/fK/lrBMeFT1xWanT5NIaCsuXfQ+dPjr8Qh8UfiBfa8IPIjmVURO4VRivM2RVPHWvbf2rPB+ieCfihcWGgBUs2hWUxI2QjHqBXhzq4bJPFcdSzd0Z+hJGvPIzV7TpDa3Ucsf3kYEVncsuAcGr+loRcIH/vc5qY/EXE+p9N/bA8T2ng86aNORmWAQRz7T8oAx/KvLvhR8UtZ+HPiiXVdORZricFXRhkNk5r7J+Gei/DpfhLZw3EWmuGtd1w020uWI5PPNfNPwEbwxZ/HCYaiIX0mOaX7P52Cgwx2E569q7ottv3bfqaJQ1OY+NHxW8R/FDV7afWoTaRwJiK2QEID/AHsHvXbfD/8Aaf8AGHhPwlBpFlZLdW9qm1JShYoPwr0b9s7UvCN54T0o6e9nJq6z/I1tt3eXg5Bx26U79m34p+BvCfw5S21i3hs7l3bzrhoxJ5vbnvTjzct+S/kQ3G1z5k034ha3Y+Pz4ojlL6vJOZmbB5JrsfjF8b/HXjTw6NO1mGS206Ygn92VV8c9xzUsuveFm/aKOo28MLeGvtgkVAMJjucY6Z5xXvP7TnxO8C618JLuysp7a8vZgotVjUZjIPUelbSc0l7t/wBBvl0PgS5Vc9ear7Q3SpbuLzGJzjmoNuzjNeVLcgs27ANgdRXtnwJ8aeONBmuLPwgk0zz4MkcabunevELdPmzmvqv9jv4seHvh7Nq0GsssL3Kr5dxjkYzkV00ebXlV2PoeZ/FDV/FWp+M/tHiQzR6pHjasgwVA9K9A1TVfi7qnw/aaaG/k0FosNJtONgHWo/2mPihonjz4h6de6Qqyw2aBJZQPv/MDzXtupftUeFv+FctZwxMbo2Xk+SF+UHbiu+Mq1rRj/wAAvSy0Plv4R3ni+38QGHwj9pa+lXDJDk8epo+LsfjGHxIo8YLcJqO3KCf+76j/AOtXa/sy/FfRvhr4p1K61VCEuYtiyADI5zS/tRfF7SPidr2lSaXCxSzUgyMOWyelS3Uv5dzT3b2sLZeEPi9r3w/N7bfbpNE8ksqGXaWjA6qpPNfNWpboZmEgw+TnNfcWh/tfaNpPw5t9MbTZBfQWfkAfwkgY6V8S+ILr+0tRublhtMsjOV9MnNcuI9rJJ1DF2fSxiSPvORUTSAfLmppVC5AqPyV289a4DMidgozTFkD/AEqVow3HakaNV4XipJIjOFbbjNPaQKuelO8sZOR1oMY70wI1k8ykabDBQKnEY4xjFG1VyMA+9MCCM7SWP1pFuPOY8Yp6r8x44p21VHy8mgCGacxMMKfwoaT5c496kZQetIyhvekBBHO0h6YApJpmRgAOtTfJnANDbep4+tAEYJ27jyajjlZmJIqWRh0FM3K30poRJHI+4d/pXqfwJ8I2Hjr4jaNo2pXH2W0uZgryZxx1x+NeXRssbDmuy+H+j6trviawtNDSR9SkkAh8okEN65ropWvqNH1T+1Z8G/BHgXw3Z3Xh6NbLUIztZVfd5g46+9aX7PHwb+H3iD4RXGueIEjvb2R5FO6UgwgdAAD1614z8afAPxD8I29u3ix5Z7eYALL5m8A+h96ufDD4I/EXxV4MudS0F2h0kkna8wQSEDnaO9ety0lFJzdu5fvFf4c+CfDOpfHy30S/mxoTXDgeY2NwAJCk/Wu+/a68I+CPDtrYL4atbWzuoztf7PwpHvXhnhnwH4j8S+PY9CsYm/tvzivLbSrDqc+ldL8bfhL4w+HX2b/hI7lbpZx8siybufSq/dczkr36L9SfeSPdvhP4V+GMfwIgvtRt7W81i4idpjLgyK+SMD0rxb4H6H4Q1X41rZa4VXRFMjxpMflZh91TU/hH9nbx1rXw8PiGyuBa6a8ZkWFpCDIPUCuD+HHw3134geNl0LS/3WpIzb2dsbAOpNL917Pr5vsJ8x6p+1lp/g231qzbwzBDa7BtkEGNre9eqeGZPhhYfs+W6/Z7O41Gaz/fswBkEh68180/Gv4UeIPhnrEFvrN0t60g4kU5APpXVaf+zD4sk+GS+Jm1CO2s5I/OjtS5+ZacnRbirOy28wjzE37LDeDIfiZqB8TLDJbLGxtFmwV3Z/wqp+1Pq3hbVPGcM+hW8NtHna/kABSBjniuZ+B/wV1X4t+KbjT7G4WzNqhklmc/d5xVP4x/Cu8+HHir+zLm+TUAxwsgPehShq7Xl26CfM1ufRHiDx58N0+BNppOnWlqblrVRIzIu8Pt5OevWvGf2YfG3hDwX4q1O/8AE9styywn7LvTcM59Kl8Rfsyax4d+Htt4hmv4x5ieaLVjjIIzxWT8BPgW3xev9Ra4vv7N0+zTMk+MknsKT5PdSjoPlmnZsp/FnxdoXib4jRahaQJHYtMvmrGmAV3DPA9s17j8ePjN4F1z4b22jaFFasqwKuY0UHpzXzz4u+F40P4hR+H7a8S5inlWOObtycc16h8cf2X9P+F/g+3vYtca7v2jDSRMBtB7gYqm/ebcPQNe58s3UYklZui54qFhkYqSbf5hGMjvUbMVU4HNeG9Xcglt4wvTrX1V8H/2htD8C/B2Tw+bJBqLM2+QqMvn3r5Tt93G7gV9X/DL9n/wdq/wdPiXWNQJv5lJjhD424r0MJvfluF/Ox5T4L+Iln4d+Llv4mngElsk5kKY9sCui/aD+NVn8VL1J4IlUDHQe/8A+qs/4M/DPSPGfxUGialeeXpcRZjITgsoPStH9pDwH4W8H64kXhxz5I+UgtnPvXpRlNRl7plKN9bnVaT+0zZWPwesfDEdqsUkMBjfavVs4zXnfwR+Mkfwt8bX2tPb+aLiJlGRnYSc5/WvS1+DPgXTfgzaaxcXn2jWLmHecNynHYVzP7MPgfwd4m8SaxJ4omVoLGPMMDNgSGiTn7qUdPzHFqOtzhvi58UG+IviFb5YsBWLfMK7/wAaftQSeKPAtjoUdt5P2e3WFgMY4GK434uaL4ctPiEsGkAQWEkoV1RsqoJ65r1b4yeBfh54c+HdoulRxvqfkBjOpBcsRUP2jlJ21No2abuec/BP9oG5+EOm63ZxQ711DHIxxxiuQ1b4iXWpeNrXxAUDyW84lCt9c17f+zNpHw+XwDrWoeIoILzVt5jjWfB2DHHBrym3s/DN78YLaCUJBoclyN6g/KBnp9Kxlz+zStbX7yFa+5p/Fb4+XvxJsVgljMYAAwx4qPwX8fNW8J+A5PDMEbfZt7MHXtnnn866/wDaQXwT5cMfh60ggaMBVeHGW4q/4B1XwLpXwdEc0EM+sTA+Y8gBYHpiuiPtpVHtdIGo9zxjwd8Rr7w344XXraMyT5bCjrzS/Ez4jan461JZ7wMjBskOOa634H33hWx+I08+sxxtaqGMSSD5c1U+NuteH9W8WRT2EMaW4f5vLAAK5rGUansm+ZWuC5b7jNT+OHibUPBttpU0Mn2WGLyxJtO0ivGp5PMkZ26sSTX1r8RviV4I/wCFa2emaRa2/m/ZwG2qM7iOtfI9wvmSsegzniuDHc/uqTuWuX7JFj5s1OjBlxUaxgZFPSFV6GvFkMeoAbr3qTYuMk1H5Y7HmpGXKYrMpEbMuK8u+IkvmapcDsu1fyFepfZ9zArXkHjabztSuj38wj8uP6VrS3MKmxyVxwuD612PwXj0OT4jaR/wkSs+kq7NMqvsJwpIGfriuNuO3pW94B8NxeKvEEVlO7JCUZ2ZOvAq60kotsqmm2kj6q8TeOPhLoNjIdO8O2styPu+fOWH1614z4t+LVjJIyaXpukW4wPmig3dvU10Fj8CfD3BkNxJ/wADxXR6f8FvCkOM6f5h9Xc14f1qhT2R6XsKkup893/jO+1BXBnVQf4YYgv8hWRNLJdQ5Et1JPnldpx+FfYNn8OfDNoq+Xo1sCO7LmtaLQdMtV/dafbRgDjbEKX9ow6RKWEe7Z8LzRXMalHWUDOdrZ61WaORTwrAfSu4+Llxu8Y3vl/Ku8gBeO9cOZn/ALxz9a9eEuaKkcEo8raHLcXEPKyMv41bj1iZhtlVZR7rzVZZpP7xNSLcf31VvqMVTESt5Mh3FcE9qKiaRG5AxRSGehW+cjNa1mxVhWPCdpHcVq2p+YVlI60fYX7Kl55mk3KbhncDivpDtXyj+ybdEXF5ET2BH0r6rz8or4nGJ+3k2ezS+BC5oLcUyhmriNAkOQR7V4F4wcWfxs0d/u+ZGyk/jXvTGvBfiuv2X4meG7jH3pCma6KPxWE9j5s/aYsGsvjDq5yp8/y5eO2Vx/SvLGTd0r3D9rO2SP4oLKPvTWMRP4Zrw4Mfxr9Dy+XNh4PyPlMQrVZINpGAeK6r4dyGPXFAOMjFcpuZuvJrofBLsuvWxB/ir0nsznj8SPaNokjGafGojXrUAZ9ny80+JmYfNwa8xo7ybYrYOOal2joarHzN3y9KmGSPepAkWNV5HFSqqkciqsIfdzU2C3TrWiA6Dw7qk+ialb3lsdssLBhj2r6N1b9rLxBqfgefRZLLEs8Pk/agOcEYr5t8M+V/aVt9q5hEi7/pnmv0K0tfhlq3wsaE2+m+Qtod6MqiUMF6565zXoU3aOsbl2XU+QPhX8TdY+HOrfaNMJkzw0fY1q/Gr4waz8ULzT21K2W0W0B2RqpHJ781s/s36n4a0f4kSzawIntlZltzNggc8HmvTv2xrnwtfeHdJutMazk1MzY8y3xuMeOc4rpcmpK8fmDtzWPPPh7+0J4x8N6D9hsY2uraFcKPLLbf0ry//hK9Vk8Wza8sjpqM8xlZh13E19Z/sy+OvB1j4ChsLs2tndrkytMBmSvCPGmteHLH49XF5ZRRvoy3avsQfIfXH407tzty2CLje1hvxM+LHjrxJ4Ri03WYLiLTnKkyPEVD46c1gfCHxx4v8L3ksXhlJ52flo41LD8q+q/i58UPA/iD4SapAstvJM9viKHaNyvjjFeJ/sn/ABQ8P+A7i+j1YLG9xjEzDJHtU3nr7tn2HeNtjyf4leKPEniHxc974hE0OoqAAkqlSoHoK7+Dx98Uv+EGeC2hvm0hoirTLCxATHr9KtftZfEbw54v8ZaPcaNsnNvH+/kUABucgGvbvBf7THguP4dxWl0y28sNr5Rg2jDYXFHNUtpG4uaKWx8ceA77XrPxHC+gNOdSZsKsIJYmuj+M174/vprIeN4buJVB8j7QhUHjn61P8DfibpvgX4rTazdRA2ckkhVSBhQzHBH4V6r+1d8dPCvxC8E2OnaTJ9ovFuBKSR9wY5GaG59tBc0eY85+Fdr8ULzQpz4QivpbJBtbychPoOetec3Npra+LHiuBcLrZuPmVsiTzM9Mdc5r6b/Zv/aY8MeCfAsWi6vm2eAlg8YznPrXivjT4s6fqXx4m8W6fGGslukkQOuM7e+KfNUk9Vp0EpLU1viP4E+KWm+EUvvEUN4+kYBbdJvCehYZyPxrm/hD4Z8ZeLdUmsvCUkiyqu+XD7QB9a+gPi3+1l4c8XfCvU9ItonOo3kPlFSMqM9TXz/8FPjE/wALby6nhfyriQfLJjOf9kj0qoupJ2k0mLmjbY5z4reFdf8AB/iKay8SQvHqR+cvI27cvYg9xXn0ky7iK9T+Pfxhf4x+IrTUpLZbZ4YPJYL/ABc9a8qZVPbmvPqfE7k3uCuE5q5Zv5kgA6k1TRQzYPSrtqqpICtRF6jjufTHgf8AZo8b+MPBMWrWl8tvbyxlooZJSC6/SvOvAvw31XxN4/j8NQ/6PfiYxOzHGwg8mvV/h7+11qfhPwbb6QdN+0C3i8uOQ9uOK8s8K/E7UvD/AMRJfE8SeZeSzNKyrx1PSvR/eK3M/Q3TbvoegfHL9mrVvhfocGtS6ouq2e8Ru2WDRsenB7VW+DX7Mer/ABY0SXUlv49NsVbbHJIpbe3sPT3o+M/7R+t/Ezw/FpFxZLZ228O/HLkdKh+FH7RviP4a6AdJsbdbi2B3JuXcB60uWo9HL3g9622pxWsfCnVNF+J//CHyOj3/AJwjEin5Tno30r1X4vfsjz+BPh7Pr6619qmtVV5rdl2r/wABOa8f1r4h6trnxCPiaYn+0DKJAV7YPArtPit8bPGPjrwjFYapbTQaeSG87aQrfj3rW0uXSdu5L5ux893Mu33qJeeaszKF6jNV39uK8qW5iLDIfMx0r6S/ZT+CWm/FfUr+TVZ2jtLRATHGcM5PvXzfFjcBnmvV/gn4i8X6HrEkfhNZpbqUZaKJS3411UVzaJ2ZSv0PRP2kvgvpfwr8SaVFpVy72t+DiOU5aMggcn0r22z/AGU/Bs3wvS5lmlOoNZ+ebtZBtDbc4x6V8r/FrUvGGpeJEk8WR3EN5t/drMpXH0ruLe++K03w/Escd+2hmLbvUHGzHX6V0yguVLn/AOCa+/pYd+zj8IdN+I3je9tNVlP2GzBZkQgNJg4AHoK1f2pPg54d+GeqaTNojPCl6SskDsG2475ryv4bah4ntvEZi8MtO2pTfLttwSxq38Wo/Gq6xAPGMV1Hc7cxicHBHqKnljdSctOxr7179D6z8E/s9+ALj4R21xfWMVzdXFp50l9JKQyMRnjnAxX5+eLbJNP1q9toG3xxTMiN6gHg19C+HfAPxZ1/4fm805bo6MIjsQybS6AfwrnmvnPWFltbyVLgMsysQ6v1B75rCtGMdpXOd31uYjKw+91qFlkbnPFWnbzPeoWkUfLXEZELbuinmkWNv4jUrYXmkWTf0pCI/s7u2QeKlMDyKAODT1kCsFr0r4M/Cu7+LHiy20e0dYmkyzyN0VR1NUkWo3PN1s5EXB5o+wOxzyBX0T8dv2aLr4Ow2N218l/a3LbMqNpVq9G+Gv7FsPi7wHa6xf6t9inuo/NjjWLeAvbJzWvKuXmew+RLqfF5tTyO9NjsnQEnmvTF+G7SfEs+F4rmOVvtf2VZV+6Tuxmvbvjh+yRp3wv8DprNrrLXNxGQJoZUCg+u3n+daey1UerJ5Uup8hyWDyNkZAHWo3hZRjr719dfs0/s06H8WPDuo6rq99JGkMvkpHAQCDjOTmvC/jZ4Ft/h3441HR7S5+2W8LfJJxkj3qJU97dAcVbQ8yW38rJ6k02SISYJOKlDFmORUMjvuwoyKxIFaPjFNWJY84p247c96jVmOdwoQiRIgx3Mea9I+CvxF/4Vj4/0vXfK85Ld/mX2Iwa8yZn3ADpU25lU4PNaRlysD6j/AGhP2nLf4sadFZWsRihU521o/DH9r7/hDPhkvhmS2AmhDJHMD1U18nRSOoyxoMrtJ1wK7vrKuvcVl0FdnsPg342XHg/4oDxbAglkMjFlfup4rV+On7QkvxclTehRFwQvpXhEszbcDrQrMq/MeapYtq7srvqSfRnh39qzVdH+HEHhdxmKBDGjDqVrz/4efGDUPh944bxFZttmfcGB7huteYfNvJLcUk26VeDzVfXHZRSVvzI5dbs9S+LHxivvidqS3VySNpz1reh/aU17/hBLfw3NI0sEMflIc9F9K8SUkLjOTinQIQwJNH1ybnz2HyrY9O+FvxM1/wAD69PfaIXaW4QrLGgJLCqvj7xlqPjDVhdXyPHdKeQ+cg16T+yJ418L+DfEGpz6/bxSzNDi3aZQQD361hfGvxVo2v8AxATUbOCJbbzAZFiUAEZ9q7qbqypNrRdw92/mVdf+KXi3U/B9vYakkxslQRxyuhCkAVzvw38ZeJvC91ex6As04nT97DEpbPvgV9EfGT41eEfE3w1s9H0yyhijjhVQpQcHbXB/sz/FrQvhtb62L20je+uFxDMwBwPStGqzlF7Mp8sTxjVtc1bUvEgupGkTUUkBVed6tngfWuk+I2teObjTYn8SW11BDKP3bzKwBGPer+o+PNNX4vWniGSCN7WK5WSRFUYIB9K9C/aO/aH0n4n6OLKyt1SFFAUY/Ws5Koub3tOpS5LanypI4VutQMQMknC1M8alizVAy7hgnivn2IdHMDXQWPjzVtP0v+zYr2RbQEkR7uBmudMajAAxSbR+NXTqzpu8HYTSe5qWfiO80y9W7tbh4Zx/GhwaNW8R3uty+Zd3DSt/tGsorz7U2TnOK0+sVOXlvoHKjWbxZqT2q2ZvZTboNqoW4A9qp2us3OmTGS1uJIZDwWQkZFUhgdBzSfdbOM1Lr1LpuQuVFy51Ke7kMssjM5/iJp82vX17EsVxcyyxr91WckCqTYZcYxQCGTbwKlVJ3eo7E8OqTw5SOWRFYchWIFdJ4B8Iah8QPE1ppOnkfarh8Bm6D3rlI2A+tdX4B8Vah4L8QQanpa77pOi4Jz9MVpCTlJKTKVkdh8WvhBq3wxvYrbUL+O9LcBlJ4Pp1rorf9m3VG+HcfiabUkt1mTfHb+orhPH3xB1rxtqG/VleOZTna4Of1rYm+JnitvB1vYNHOdNhTYkuDtxXox9mpO7Lur6IZ8IPg7e/E/xBc2UVytpHbAtLKTz+FUvHHwxk8O+NIdEW9W586QRpLn1OOap/D7xh4j8M313JoaSzSXC4kWNST+lZmua/q+peIvtV15iagjghWX5g1Q/Z8iJ+R6z8Vf2c7T4f+E4LyPV/tF60YZ4/4enQV88NncRjpXp3jzxN4v1DTYRrMNxDbsoCtIpAIxXmLN8x5wa4sb7NSSgPXqrDdxHbmpPm2jIqIybTzVhWDL1rypAMic854pzOWwBTCeuDTGk2sOMfSsx3LsIPy7v0rxTxhMkl9OUUjMznn6mvaYZWRd/GccZrwzxLJuvpMdN7d8962orUxqGDNnivQPgfD5nimd8fctz+pArz6bqMV6d8DYv+Jhfynsipn8c1li/4UjfD/wARHvdq3yitW2b+VYVrLt9xV1dYs7dh5t1FF/vOBXyUot7I91M6FT8opJn2xSH/AGSayB4s0gYH9pWxY9vMFWby8ik0+eSKVZF8snKHPaslGSexV0fG3xCkM3ii+frmRj+tcua2/F0hl127Ykk7z/OsUV9lT0gjwJ/EwU0panbPehl4HFaEjKKlFs7DPSigD0OEGtK1b7orLibmtKzYcZNYyO1H0l+yrdY164QcZQD9a+v0+6K+Jv2Z7wW/i+NCcbwa+1oWJjWvjserVj1qDvAlpvHNBNNLV550CMcGvDPjp/o+uaDdZxsuQPzr3BjXiX7RERW10yfsl2hJ/GtaXxol7HiX7X0ZHibRJxyJLIf+hV8+fWvo79rVfMj8JzhM+Zasob6EH+tfOhAIJr77LNMNFHzOLX71jRnHPBrW8KyeXrdqenzishfvGtHQZBHqkDHoHH869focPU93TbFEM46ZzSxuJFypzTIl8yAAjgikjhEH3eleYzvLKyBTgnFSrg9elV1hEnJ5NWVUAYxUjE85N2M81MGGM+lQ/Z1Le9TRxhuvIq0Ui/p4+0OiJ94nAr6Y8P8A7LHjfXPBkep29/DEJYt6WbyEM64+mBXzPYsIJY3XqGB/I19X+BP2urnw34Ui06ay+1SQpsSQnpxxXfR53H3HqVzPojxnwJ8N9V8WeNl8Owf6LfLIUkZyQIyDzmu++NH7OviD4c6HHq91qKanYqwRihbMZPTg9q4PQfiRqOi+PpvEkB23E0zSsPqc4r0D4uftI6p8R/CC6JLarbxu6u7gEFsV2NT0s9Cm30QvwR/Zy1D4maJPqI1T+zbUHYm0biT9PSvPfH3w01LwT4+bw3dypNcl1CTKcBg3Q10vwn+OWvfDm2e2sgZ4evl4yPyrlfiJ8RNR8ceNjr9yPLufl2gDoF7VT5r6y0Er32PaPFH7IN7pvgGbWo9Z8+7gt/tD27D5SAMkA18oXc7wSNtyCD24r6J1P9pDxbceBJ9JMMhtpYfJNwyngYx1r5yuXBYljk55rkqOUd5XE79Ss148nLE5qCS+kVsAnH1okPJqJmFc3tJdzFjmujjPSokunkb5s01j+VR+YOgqed9xE8lxJHnZzRHcM0fPBqEzBevFIrbskdKOZgPW4lDc9Kc0zFfl61D5y5xnBpfMCqSTijmYD9zY+brUTb93HSnxyCTpTXkCsAakBeeMdatWKsrruPeoEbnirVu+9hiqW5Udz9Af2cPD/wAO5/hfaNfwafNfSqTcG6ALfr0FeDaPp/hOw/aHltmMcnhyO8baCcpjPT6Zo+D/AMA/GPxE8NtqWlXK2NljajSylRIR2FcFP4A1qy+IJ8MzRmPVhc+QRu/iJ659K7kqd9GbRvd3Z9YftUXPgW6+F7mxXTzqSun2Y26qHA79O2KxP2Sb7wNY+ELp9XWxXVHlO57pVJ2joBnpXnfxb/Zf8SeAfB51+61WHULaHb5sKlsx5789eay/gX+z3qvxasru7t9RTTLOFthlYFizegANUlR5Wr6dx68ujF+KuoeFT8foptPig/sPz42nWIARtg847V9IfGL4k/D68+EOpWkctpN5lrsggRRlGxxj0xXyJ8QvhPqPgL4hDw1cXEdzcOyCOdeA4bofavY/Gv7HM+j/AA8uNaXXDNfW1v58tuy/IRjJANKXsdL38iZXdrs+PL6MO5I4BNVvL28Hmrl8DG7KB0NVtxYHPWuKW5k9xqQ8g7sGvpz9j/4naD8OdW1R9YXY08arHNjJGOor5jjkbd04r379l74R2fxY8Sz2+oXBgsrWPe4U4Zs9hWtK1/f2KW251n7Vnxe0P4ha9ox0lfNWzU+ZIRjdk9K9X0/9qbwzY/DWPTRbsbpLPyfKAG3O3FePftOfA/SvhVqmkyaRcNJb3oIMUpyykH+VezeEf2X/AAZqHwrt72+eR9RntPONysmAjbc9PSuvmpKKbjp0K0srs8D/AGe/ilpfw78eXmrahAphlVlXHG3LVsftOfGzTvihfaWumxFYrPJLEgk5rB+BPwv03x38UJdG1G4K2NuWZypALgHAANdn+1R8GvC/w2h0y60KRo3nYpLBJLu6dDVXgmvd940+1vqdr4V/a+0jQfh3Z6a2nSG8t7XyVx90kDFfF/irUjrmuXt86hDcStJtHbJzX3l8K/gR8Pb74SWt5qlvDeXV1bebNcvLhoyRnC88Yr4Q8bWNvp/iPUbWzffbxTMkbeqg8VzVeVr3Y2MbLWzOekjXJwOPaq+1efWnMHDEMajEblic8VxmY7AbgmjavYUxlLcDinJG0fU5oAngUDqOtel/Bfx9f/DXxZDqumr5koUp5fXOa8yjiZmznFe4/st3mh6d8TLGbX0jezRGI80ZUN2JFdFG/NorlRt1NL41fGjXfic1nDqULW0UJ3Km3aM11nhP9oDx3pngmPTrK2klsoE8oSLEWwuP73at/wDa68WeEvEVvpC6ILdruGQ73gQD5fTivVfhr8YPh/4S+GFhZJPCJ47f97DsG53xyTXrRlU5bqnfy7Fy5bI+F7bX9RtfF41aN2F+tx5w4/izmvSvip8SfHPijQ4YvENrcw20uCkkkTKrD2Jrl7HxLpv/AAuBNXmhU6b/AGj5zR7eNm7PSvo79pP44eGvGHw/Ol6UI5T8pUsnK47D0q4uom1Fer7EOzPCPhHffECzs9Rg8HwXl1buN08dujMo9/Y15j46uNUuteuZNaV49QDESLICCD+NfTX7L37QGg/DPwzqunarFsnkl82ORR94Yxg14J8dPF9v468eahqltEscUzZG2uas6ns2n8P5kt9LHmUjA5xzVZ3VTjODU7Rhc8VA9uGYMa8ckRsdTyKFZGHFLJHkYoWMRqMCgQxpEXA705mAXngU3yRuzQyhvpTAd5wdRjmozcKrbcnNO2begppjXcDiquSDyBVJPNKr+YuRS8Nw2KTgcLxRcBizlpMY6U9pPLHSnKqr2H1pXA4JouA1Dlcnilgm3SY7UvDZxSxKoPHXNWgPcP2dvgO3xkutSZtTXTrayUF3zySfSud+IXw5bwr46XQor1bpJJRHHMfc45qv8Ktb8V6TeXKeGIridpVxNHApPHqcVkeJtS1fUPEDPqAliv43xtYHeGzXrwUPZb69iteqPa/ix+zLYfDvwTbaguuC71BoxI8ORt6dBVX9nr4BaJ8StB1XWdb1Q2cFqSixRsAxOPeuI8bXnjxvD9s3iC3vIrCQARzSoQCtU/h3p/ja8s7/AP4Ri3up7Ucz+UDt/GtbU+ZLmfqO7b0RPB8ObO++LUHhmO//ANBmuNi3DDnbnv713X7SHwT8J/DbT7c6FfSXFyo/eln3K3rivHLWHXb3xbFDbpO2s+dtWNR84f0rd+LHhnxr4dSAeKbaeDzRuQyc5+uKyl7Plle/kNXtseUSb2c46CoZN3AXqameQbjjrUTyeXya8VkCbGXhuDTEjk3Fs8U8NuGaYlwGbFIQ197sFU4zSMrRnaTk0SziNhxzSGTdkkUDGKjLk5pDGzsOcVKkm5elRtJtfGKQh204xmlWHb0bOaQyEL0zQrEKMgimA6G3zISxNewfs56/4d8N+NftniGGOW3jjJj8wZG6vIIZGJPpXp3wV+FjfFHxA9obtbOCJd8kh9K68Lf2isrjs5OyN/47eNdG8WeKo7vTYIUhVslo1wCvpXf+Lvi94YuvhnaaLp1pCrCAK5VRknFeT/ED4bQeEfFiaVbX32qCRwm9sZXnHNeiePvgJoPhDwZbXdrqRn1Fog78/LyOmK9mLk5Tah/wCWmtGzn/AIE/EfRPAsertfW8bTyjETOM4HtXMal420+6+JFtrUkKvAkwZlxwRmuw+DPwW0bxtoOo6prF80CQZVIlbBJ9a5LR/h7Yah8UIfD73m2xaXmZuPl9M1m3U5I+7oP5nV/HL4zaf4502O1s4kWJAAAo6Yr5+8td2T3r3j4/fDfwx4PjiGhTFpF4b5shvevB2VipIryse5up76syvmMKKakXA4FRMGbgcGncr1ryWMlCDacCm+WOpHFLGDg5OaXaW71IENxlbeUg5AQkflXhutPuuz69a93uIQdPuzn7sLH9K8E1Ri91zXTS2MJ6szpfvV0XhHxdd+GIrhbRIy8xGXcZxiudl++afHMsa80qkVJWZpB8rujrdR8fa5qXEt/Iq/3YjtH6VhzXk0xzJNI5/wBpiaoi4DHA5roLbwbrF5ppv4rGV7QDPmBeK5VCMdkdPM31MlGDEd/rWxpfjnVvDQeOzvHMLKVaCUll59PSuemkaF2XGGFQrlmLPkD1xVuEZKzRHO1saV54f1HVLN9WWDfbucs684Pv6VgrC27GOa39P8TTaZY3FpE2bebkofX1qjFcR+Zu8rdSjdaMHZq47S9BvdYuo7aytZbmdzhY4lLGulvfBlp4TJOvz4u9vy2NuwZ8/wC0RwKpQ+LtUtrfyLKb7BGRgi1UIx+pHJrPeB5m825kOW5LOcsadxWKNzcGWZmijEadl64oq211bxHakKuv95+pop84+U6eMjj1zWha/ez+NZ0fUelX7U4YGokdKPY/2f7wW/jqzU9WOK+67Vt0CH2r4A+Ddz9l8badIOpcL+dffGnNutIz1yM18lma/eJnqYf4WXc0hNNzSZ4ryNTqEY8149+0bHu8J+YP4JkOf+BV6+1eX/Hu3M/gW9IHMYDfkQa0pv30J7Hif7TsJuPh94OvBk+W0iHjI+ZF7/ga+Zmwy19Q/G/OofAXQZyjOY7qMllP3cow/LpXzDLhYxjge1feZVK9Brs2fO47+IQ8r+VWdPbbdRd+e1VNuT1qe1bZMvfmvcTPMPfrKTzLWM/7IqSJmYkEVR0KQSabauOhjU/pWvGwbkV5stzvjsRsrKAVHNSRsWXkc1LtCjPamgg9DUFDN0m7/ZqcZb7vWmeYoODUq4j5NWgLenhjIA/AzX3v8A/B3w51n4cwxXljYXN26f6Q93gtn2J6fhXwRbSqxx1r3n4T/Bnx5470Jr3RZfstkOFaaUoHPoMV2UuVxak7FWdtHYy30jw5pvxtuNO3eZoEF8UGTkFQema+i/j/AKR8Pbj4Tz3GnxWEF9Cqm1a3UK+c9OK+UdW8Fa7ovjZ9AvLeRNZEwQoTnJPQg9weua9T8dfsy+NNB8Fy6xc3sV5BBH5s1qrkui9z0wcV1+40rst301Og/ZK1LwfaxXya1Fb/AG5zhZbgAjHoM1xn7TX/AAjMfxWt30NYfsjRRm4W3ACFs8n8qx/gn8HdX+KF9dQ2F0tjFbgGSZifyAFUfjR8H9X+FfiCC21OdbxLld0NypPzYPI57iiXs+bfUaT5tGfVFl8QPhtN8K5rWSG0ULZFWtzGMltvr9a/P3Uo43vJzGCIi7FR7Z4r6n8J/si6x4q8Cx6qdYW2ubiLfFbYyCOwJr5d8QWM2jald2U3E1vI0b/7wODXNV5PsEPZ6mEY/L3DOeaikj8znOKk84tkEVFJLt7ZNcpmxpX5cVEkO1s9am3bl9KiWQ7sYwKRIrReZjd0pgj8vhTSySMPu0iszfe4oAPs6ded1K0e4YPSmbpNxx0qTJ24HJoARI/L6dKcyAjnqaaiv3pGWTeMUCJlXBFW4QO3Bqn2461Naq6t8x4zVoo+nvgz+1JffDrwmmjiyFxDECVb0rzrUfihqGrfE5/Frri9efzgo6DHQV7/APskeH/Ad14Lnutdhs59SkdlK3eCAo9Aa8m+Imn+FbP49Na6YI18PG5jLiM5Ud2A9q9OL1Vo/M0jy3d2bfxU/aT8R+OvBp0W5tRb202PMkxjcByKwPg/8dPEXw1sZ7HS0aeGRt/lhN/Privov48+JPh5J8Fb2009LFrgRqLZYUUOjjoc151+x74m8I+HYdVbXDbxahK4EUk6g4UDoM9K0vLfk+RUeWx4h42+IeseMvGza/fNi+V1IXbgrjpXoniX9ojxzrXgmbTZ0mWxlj8prjYcFcf3qrftJa94Z1j4uJc6HHF9j2x/aGiUKHbPJwPavo7xJ8VPhzH8IbmxVrUq1iYktljGQxXA/Ws3KfSP/AFLl0Pzwu8bsnk5qm2Gbirl9H5kzemaqrDtyCciuCWrIYR7c16P8I7/AMS2mvLH4Y85r2b5fLizlhXnUduM5zXvn7LPxF0r4beLri+1WMMjw+WrYGV5rajzc147j6amL8X28ayatbjxhFcQTAfuRNkD8K7jw9oXxZ1z4f8A2vTlvJNG8ohdrYygHOB1NWf2qvjRpXxMm0m30yP5LMszSN1Oe1eleDf2svD+ifDuy097UrdW9t5IVR8uQuK7eaveyWpV42VkfJ3hSbXbHxRFDpHnDVZpPKCpncWJ6V1Pxe8N+OdHu7STxjBcIZQTA0jbkP0OetQeBfHWm6d8QNQ1jUrbMN0ZGTbwYnJyCp7EVu/Gn45XfxJ0XTNLuQtx9gkLLdBMM4xgZ9+lX7/Jfm0NHa5r+CPhH8TvFXgP7fpTsulshMUbTbS6j+6v518869BNYahcQ3Kss8blXU9QQea+s/AP7Wx8JfD210X7BvmtYfKSTHtXyn4r1Rtc1q91CQYe4laRh9TmufEe05VzvToZaWOdaQSGmNMFbGKmkVc8cVHtWvMMyN5OM4pUkMnbBpdgY+1ObBx2FMRJDLtYDGfevV/gT8O5vih40ttGguPsu8F3m/uqB2ryqEqDg16H8JPFmqeDfFMGoaMHa8AKKqDJbPauvD/Fo7Fxvc9g/aE/Z8Hwpt9NvI9VbUYrp/LKyDDKR3r0jwH+x7peu+A7LVL3V57a9uoPOHlKCqgjIBya8L+L3xI8WeMprWHxDFcW4jO+NZkK9uoFdX4b+KHxOk8DpBptveXGlwxeV5sMTMuAPUV6HJJaKpr1Zq+Z9DyvTfBIvviYvhlJ8K179l849B82M4r3n4+fsy+Hfhv4FTVdP1ed7yPaJIbggiXPcYHFfNtjqGpReKFvoZHXU1n3qQDu35/xr034pXXxIn8Pxy+J7C9gsZ8FZrhCFPpRFLdzsl+Jk732O7/Ze+A/hP4l+GdT1TXpnZ4ZfKWGOUJsG0Hca+fvjV4XsPBfjfUtN0y4+1WcTkRuTk49K7X4ReDviDrtnqEvg6O4a3Xico+1ScdOTycV5d42t9Tsdcu4NYjkiv43KypKPmBrCpycjfNd9uxMr9TkNzFju6VBI0nmAL0qyzh84NRNIqcd68tkET7vTmmxs+PmqyzBY8mofMEnK9KQELLIz8dKdIrbcL1pzThW205iV5IzVIQyNWVMNUYjfzM5+WrcZ8xPSmFvm24qiSKWIyL8vFIsJTjqankYKoOOaM5X3xTsBXjhYOSWz7Zp0imbjOKI3ZnIxwKkYlegoGMVNq4zT4YfLbJOaUZHJGafDuk5IxiqQH0b+y/8atJ+Fem63De2sb3V1gxTOoJAxgiuD8ZfEKz1n4nwa+tuskEVwsjqF4fBzXo/7MfwR8KfELw9rGr+Jr4wpaNsjhRwCflzmvMPE3hHTNN+Jv8AZFpKW017gIsjn+Emvapv93aMequwtre56r8df2irH4j+G4rC3hSOJEAWML0/wrM+Cf7Q0Hw38B6hoHkKk1xIzibAywI6ZqT44fCjwr4T0GC40edvtaqpcbsh8/jU3wd+GPgbWPhvea3r120+plmRLZXwEx34rocZe1jFQXoVy20cjzjw/wDFBvD3xSi8VJbJK0cxcxkAg57/AFra+PPxzl+K0Y3oM5z9B+VU/hf4T8P6x8Y7fSNVn/4kgkclmONwHQV0X7UnhnwZpF9CfCkSQInyt5TcNx1Irnm58svd+YuVWvc+cpI1GcioWjUnnpUzxsSTnFV5lZiADivCe5I5gO35VGqhc9jUiQMRjNH2RxyW/CkUQFRuBPJ96RgO1SfY2kk64pz2xC4oAhyu3H8qZx681OtptHXNNWzyxOaAImwrZIAp2fM6tUjWhbqcCla1C9OBQKwkLBQR3rt/hzdeJLO/lHhxJpJpFw6wjt71xsdsFyO/rXs3wD+J9l8NJNS+1W6yvNHiN3GcHvXZhNai1sNWvqcXr8ms3mtKuoRzR3wYDa6/MWrqvF9n4307QYJNZtZ4bSRQFZ+uPemeJviDa6l48s9XEausM6yEAdcHvXafGD47R+OdNW2VFC7AoCivXjF2m+ciXLfY4PwRpnjG60e9k0aOZrFRmQjpn2rntB0fxD4i8XJZ2KSHVS/HPKkV6N8PfjY3hfwXcaN5WGZid2OoPSuZ8H/EiTwv45bXY4gXbcDx0z3rNwi1Bc+g/dS0WpifFDwz4m8NXKRa+dzMOGDZH0rz9X616T8VviPP46u/MkBAzk7hivOOMYxXkY7l9q1F3Q1tsRlsfWkDf3jxUi7aZtBNeWyiZtqj5TuFM8w5pVXgUcDj1pARanIF0HVTnJ+zEfqK8Fvv+PkkV7h4izF4X1Nx02KpP1YV4fef8fD10UtjCXxFKRizHvVWZvm4q2w5JqnOPnNORogSYriuy0b4pa5pOktpkdyTZt/A3auKWpQKyaRZcurhrmdpT1Y5NWLNluFeNvSqK8j3qxbKVlBHNAFcW/74qWCj1Y1oKbW1j+Z/Mb0QVei0l75GuWQLCnVsVkXiAzNtGF7Vk5LY1SJf7WVOIoVjH948mq73nnNlySfenW+m3N5gQW8k3/XNC38q3tO+FvinViv2XQr2QHofKKj9azcordj5Wc9lD3or0a3/AGb/AB3cRB/7J2Z/heQA0VHtqf8AMVyS7GWmTV23+ZgM1STr7VbtztYEV0SLidv4Aums/E1g69VkX+dfoXoEhk0uA/7A/lX5x+Gbjy9VtZM4IcY/MV+hngm6+1eH7OTOS0anP4V8zmqXus9PDbM6GlpgOTSlq8FHYI2OtcD8Yo/N8D6ovfyGP5DNd2x/GuR+I8f2jwrfoRnMLD9K0h8SZPQ8A8Z7tQ/ZvDo27yZIj+Abb/X9K+XpfmTIzg+tfTe37f8As463ETzbjf8A98SA4/Ja+ZbjIXGPxr7fKn+7lHzPAx695PyKuDmprf8A1yH3qLripE+WQE17yPJZ7d4VcTaHZnORsArajg8tsgn3rnvA7f8AEkth1PIroYZm8zaQcV58/iO+OyLSxll5PFKIfLGB0pDuVeOfanQyFlO4YIqCxDbjk96eFz1qORpN3yjipFyyj1qkBZto9r5FfS/wR/aYufhzoX9m3Fut1bJ91WGMfTFfMlqzrJgjrX2B+yT4F8EeJNHu5tdtbe91MvhIrnG1V6ce/Su2i1Z3Vymk1qeQfET4rXHjD4oP4pSL7O4ZPLVewUYFeo+Jf2sNR1rwHd6I1ovn3FuYGm25OCMHNcr+0R4N8NeE/izBY6KypYTJHJPCj7liYsQwB7DH86+jpvCXwy1T4TyAWmnxqtkWEm1RMsgXrnrnNdPMkvhKtHRnyn8Ivi9qXwv1CeS0XzI5fvR9jTvjX8YdQ+K2pWE1zEIFs1IRMdCTzXQ/sujwwPG1xJ4jSKRFTEHnDK7s9a639sCx8ILb6Hc6AlrHqMkjrKbUAAoBxkDvmm5NacvzNLRuc34P/ac8T+HfC/8AZ8Nu00cUZVX2kheK+edauZNT1G6u7g7pZ5DI/wBSc190/s+654As/hzbWN1HZrdMpNwZ0BLk9eTXxv8AFSHTR8QNdGiqF0v7U/2dV6Bc9vxzWNTma1VjOy2Rwkyru4GKhdVXt9KttZMpOTnPNMexaQdcYrjsyLMpdKThuKtLZleCKj/s7awfJ5osxcrIdoTJPSk4yMc1baz8zAx9aQWPl5Aosw5WVWdNwHek4Uk1a/s3d8xFO+w546jvRYXKyoHD8CmtKI8VcaxEXaq8luu7J60hPQSPHXrU8Eys22oVjwMCrMMK8EDBq0B7p8D/AIJ+KfihYXM+jXf2KzgO1pZGKqWx0rlvG3w91nwf48k8P6kudT81VVlbIfeflYH0Oa7T4G/tB6h8KdFmsLaATQyPvIIzzXJ/ET4mX/j3x5/wkd0uy4Vk2IBjAQ5Ar0uWaSbloaJ+R6x4+/ZM17wx8O5Nfl1mG5ltoxPPaHI2g9cHua5T4B/AW/8AjBcXzxXw02xtcLJcbSSWPIAH/wBerXjD9obX/FXgZ9Imhb7NIojaUg4xjp9a534O/GHxB8NZp7fRITcNdEExqCS34YraUZJ2c9So3sJ8Zvg5d/CXxlBpFxeJfLdRiWGZRtLAnHIJ65r2qT9jWOb4bHUjrTLqn2T7SI8Dyvu525+nevAPih8Qtc+InigajrO6K7gQRrEQRsGc4weldy3xq+Ilx4FezjS5OmeV5RuFjJCr35Ax0rLldmua3cp8ztofP16DDIy4yVJH5cVUUs/augs9Futavkt7W3e4uJDgRxruJqTWvCeo+HbsW+o2M1lMy7xHMhU4/GuKcdRcjZzis6t93ivY/wBnH4b2nxO8dRabf3H2e0SMyyc8tjoBWFpPwb8U65oR1aw0e4nsVBPmKnBA649ab8PP7esfEqQ6Ckw1STMQSIHdnNXTj729h8jR7B+1T8F/D3wyj0e60SZwLpmjkhdt2MDO79K2fA/wG8Fap8G01rVNQaLU54DMMMBsbHAxXlHxg0nxxYS2TeMIrhFcEwGU5B9efWt/wT8JfiL4w8EvfaUkraUFOyNnA3gf3R3rqjyQnrK67l25oOL0fc5/4E+BdM8afE610rVpgmnKXaR84346D8a9O/aw+F/g3wRpukXXh0LBdSSNHLCsu4MuMg+xzXhPhfSNebxdHp2lRyjWDIYBGn3t2cEH/Pauo+L3w98beEXtJPFcUgjuCTHIz70JxyPrinKNN+9e3kKPNGybPpP4B/DX4cXXwjs73WoLK7vrqNnuHuWG9OTgKO3GK+JPiFY2ll4u1e2005so7l1hI6bc8V7p8N/gR4+8beBf7T0q5FrYHd5MckuwyY9BXz3rlvPY6ldQXQIuI5GSTJ53A4P61z1o07Xi7v8AIyfN1ZhNG65yc81F5bs2c4q07/N7VA85VsAVxGJGynGM0ixNGOTmnNJgFsZpnnFuMUC1JY49zZLV7L+zL4m0nwp8UNOvdYVGs4w3MgBAbHB/nXi4mKdBxViO4aP5lJB9jWtOSjK72KT7H19+1x8VPDPja10iHSBG80MpZp1AB246fSvRvAP7Sngvwj8N9O0mJNlxb2oUoq/KXxyT+NfABvJJf9Y5JHTJo+3TLlVY7fTNdirUdnG66alOo7JM9J0HxrZWfxch164RWs11EXLLjjbuzX0N+0Z+0ponjrwPJo+l7WSQqzbsE8dMV8XCU9e+c0JPI33skVMa0E3Jxu+hLlrc+oP2d/2l7T4U+FdR0e5tldpZzPHLnnJABB/KvDvi34z/AOE88YXuq8ZmbJPrXGTO+7CnioZnZkPPNROspQ5VHV9SNRu1Y+gqJoVZtxHNJEH2/N1prLI0nBwBXGApXqD0oWNccUTbtvB5psKsg+Y0kA5IVJq/Y6XLqEywQI0jscBVGSaopEzSbu2OK91/ZT8UeHPCfxLhvvEkSSW6QusLOoYCTjBx+dbU4ubstwPJdQ8P3ekzCG6tpLaT+5IpB/Wrdv4H1i40t9Tj024ksUzm4WIlBj3xX0B+1p8StB8d6vYPpkES+Vx5igBivpx/nivTtO/aG8I2XwPtNEsbaOKb7EsMsbIMb8fMce5HevSjhJuSgt+vkO8T4q0bwtf+JL5LPTrOS9upPuwwpuY4+lN17wpqHh2+ay1Kzls7pesMq7WFe9/sx/FTQfhn471e+1O3WX7VEUgk7Rndk4PbIrE/aM+Jdn8SPGkOo20asIjg7RwVz0pfVXq3sgdux5nP8K/Edp4fXWpdHuk01gCLgxkKc96r+Efh3rXjm+e10bT5r6ZF3MI1yB9TX1N4+/ag0nxL8M4tEtbVLcC1VGiA6ELjBrg/2bfjjZ/CVdeiltFeS8CtHMRypHG2reElFK7V35gpRb2PDNZ8I6loGsPpV/aSW98pCmFhzk8Cuk8RfBXxX4R0GDVtU0ma1spgCrsO3uK3PH3xIHib4kQeIiglaGaOTaw4O1s4r0n4xftKj4jeFhYLF5eUC+WBwPWrWFXM1zKy3C66I80+EfgHxx4mtb+XwtFNJbRf68q2E6dPrXK3Gg6zqHi8aY0Er6xJP5Sxj72/PSvV/gz+0Jc/DDwdqmiRQjFxN5gkUc8jBrgLP4lXWm/E608VxoHntp/O2sOvB/xreSfs0nPToTdX2ND4rfD7xv4HgtE8TI6xSL8h3bh06H8KueAfgv418R+C7nW9MPkaSM43vtDkeg71c+MvxwufihGBOrZzuAI6UnhT9oDVvD/gGPw18xtYAwQL3znk1fL+8SdTpqxSt/KcP4I8A6/4w8bx6LpiyDVtzBm3YK4OCc1b+M3wt174a6klvrV0t5I6/fVs7T6H86b4A+Kmp+A/Gj6/YjddOGVsjk7vpUHxW+J2p/EXUGm1AMr5yd/HP0rmnGKpN8xppbbU8yaTLEY4qN2xjAzVrgKc9aj+VmHFeKZlnS7B9Qvre2iHzzOsa/UnFez/ABE/ZtvPh/4Tg1W41GGWZ41keBeq5xXj2myPBeQzQnbJG4dMdiDkfrXo/jjx94q17S0TVYZktyoRXZSFPHvXpYeFGUG57j16I6r4G/s12/xQ8K6n4gvtUFjaWzFEVerMK4DS/hW2sfFGLwnDeRlZJ9i3B6bcZzWt8PdS8dr4bvrLw9BczaZv8yYwqcA965DTb3WofFUU9n5y6usuEVQd4bpiteSjZPqa3dldHoHx2+B9h8LPs4s9TF433ZVOP0rrfBf7Ofh/UvhGPE+p6p5d3cJvihRhkcd68q+Iq+KlvIR4liuIpXBaPzgcGtPTfDfj2bwWLy1iujocanHzHbjrwKpKlzvTQerd7Fz4F/CPSviN4+vNL1K+FtYWiGQvnBcZI4qt8avh3oXg7xfHZ6JdGW0kYId53bTnGfpXP/DzQ/FGta+8XhpZ2v2Q7jEcfKfX8jUfjLQfEWn+IvsWtRSfb2YBVzksSe1Ty0+TbUOb3LWPbviJ8F/Anhj4e2U9jOZtXa3Ej3Af+IjOCM4rmfgH8OvCviTT9ZvvEc2DbtthhL4B469ayfGnwv8AHfh3wtbX2rljaPGrLGZMsF7ZGPSsL4f/AA88VeLrS+udEyltB/rnZyqn26Vvy04zVokOcubYdceG9D/4Wha6erlNHlnCsS3IHcZrtPjt4d8G6TbxJ4ehjjePA3RnO6vMNO8H6zrnjOHQ4486pJLtAJPBHfNdD8Uvhfrvw9miTVbtbsMOSrE7T6EVKlBQkuXUzd73ueg+CvDPgSH4VreXyLc6vMrElm+77Vx/wVtPC8vxBuP7dKvYQozRRueG5pmm/BfxLdfD9vECXCw2Ui744Gflx14rB+E/wm1T4la3dQW1ytmtou6WZm6e1DkrwSh/wRpS3ZJ8erjQbjxIX0OFIIe6x9DXlRXauc5rr/iR4Rn8Fa5JYzXK3e3pKveuOU8c14uKf716WGNClmBzUvl4FR5O7GOKezfKMD5q4WBKyhUHOajWMZ96XLhRuBpqq7McjjtSAoeLjt8I3mTgPIij35J/pXh1xzO5969u8dTLD4R2twXuUA464Vj+FeIyfNI5x3rop/CYS+Mg2nP41BdRgc9zVtR39q+g/Dv7GGu6/odhqT6nbw/a4I7gREHKhlDAH3waxrVoUV77N6cHN6HzYkRqxHblq+oP+GINajX5dUt2P0NMj/Yt8QoTuvbc+nNcX1yj3Oj2M+x80R2pJxXe/DH4W6n8QNajtbSI+WpBlmYfJGvqTXuemfsb3sTB7u7UgHlYx1r13wv8P9T8C6atlpNpBFEOrfxMfUnua5q2Nja0NzWFF394wdO/Zu8M2+lxWl1504UfNtbbuNa2mfs/+A9MYMmg28zj+KYb/wCdbTv4rjzmzjkHqGH+NQ/214it+ZtIYjuUOa8Z1KknfmOzlRtWPhDRdMQJa6ZawAdo4gP5Vpx2cUeNqKo9hXKr4zuIWAuNOnj+i1ct/HFkzKJC0RJx8wqbvqM6NYR6D8qKzk8QWcihhcLg+4oqRWPzvQ81bhPSqqjJqxH19q+yZ56N7R5ilxER1DDFfoD8K7r7R4R05zx+5X+VfnvprMsidua+8PgbcC48E2GTu2rjNfP5nC8U2elhpatHpeeaUtTPTmgtXzR3XEc1zXjRfO0O7T1jb+VdFIeK57xL+8024BHVD/KtIiPnbwnbnUvhD4x0xW5MNwqn+6drf1r5hmYbcHk/WvqX4auPsviuxblcy5A9wa+XJ4/3pxkKOADX2WV/FNeh4ePXwsrjtT9oyBTdvSpCOn6V9Ejxmet+AZN2ixgH7rEYrro2XcPWuH+HbH+yWHOd+f0rtIbUs27dXDUXvM7qexfGMZNC7W70z2zSKmzJBrM0HtIsfBpUb05FNaESDJ5oVQoApoCzDMjSYzg1638GPAPizx1eSweGWMTRrueYybFQfWvIoolVgQOa91+APxuf4VyTL5e+GXlkPeuyhfm93cvpsYPxR8AeJvA/icW3iRSbycbo5t+8SD2P1r0ex/Zr8f6p4JXU4bpBDJB5gszKd7LjOPSuZ+Pnxm/4Wxr2m3UUAggsUIQd8kgn+Vd54T/ay1DQfCKaZJaid44vLjkbqOMV2WnunqXzabHjXw78C6x4x8Wx6NpxaG7LFXZjjZg4Oa6741fA3xF8MbOzvNUu01Cylfy1lRj8rEdCD61yvgn4hah4R8XPrdsSJ5JGkbHuc12Pxp+POo/FDQbLTbmHy44JPNYgYycVUuZrR6Du09Ebfwb/AGadT+I3hZtXXV/7NgclIlXJLY9favMtW+Emo6X8TG8H3TIb3zxGJM/KQejflXafCf4+eIPAekGwso3uIVGRHtJArhfFHj7VfEPjqbxJMxjv2kDDtjHAFQ4vrLQXvX1R7N8Uv2O4vB3gG61yy1X7RcWUXmzROuFIHXaa439nX9n20+L0t9Nf3zWdpbcARgFmap/Gnx48Ya74Hl0i7imSxuECNMyHDD61xXws8ZeJfCl86eHhNJK3PlxAkn8KSprZyGue3mXvjv8ABNPhP44tNKgu/tdreR+ZDIwwRzghhXumn/sX+Hb74dpfPqs39qva+eJUA8tW25xj07V84/FLxd4j8WeJEufEAlhvIV2qsoKso+ldhpPj/wCJTeDZUsY72XSfLKPMisVAxz2pezT0T2E+cz/2d/hNpvxF+IDaXq84isoFYyKpw0hBwFBPrXU/tV/AXQ/hUdJvdBeRYLxmjktpH3MCBkEH0ryXwJca7D4jj/sMTSajIcLHDncxPtXT/Fq18eK9mfGNvewL/wAsDcA7ffB6ZqvZwve/yD3r+R9B/A/9m7wN4n+Gttf6oftt/cxlpHWXb5GegGP618/+E/h3oc3xwk8N3t7v0eC+eIy8Deinjn3rU+Geh/EzXNBnPhhLqSwi4Lq20H2GeprzyPStch8YNaeXOmt/aCrRkfP5hPSl7OF9/wDgCvK59IftYfBvwR4V+HMer+H4YbO9imjQJE+fMUkA18XXEbbvlr334rfDf4j6D4WjvfEcUp0oEH7+4IT0yO1eEXUwXtXLUUU/ddzOV7alZQe/WpYFdWyeRmmoc8jpUkU43YrIlH1l+x7Y+CvI1G68SC2lvWIWFbkZVR36964T9pK38Nw/FqT/AIR0RR6dJGhlWAYUPnkis74E/BvXfizcXSaTcizgtx+9mkYgAnoPeqPxa+FWr/DHxYulanIt1LMgkjmRiQ+Tiu73LK25vG+59Rx+JPhovwGbSJBZPK1kymNY/n8zHDZ65Br5p+APjzT/AIc/Ei31S+tRfW+1oArY43cbhnvXe/8ADJPiab4ejXxf24b7P9oNk7HO0DPXpnFcH8Ffg7efFzxJJpsU6WcMCb5pn52j0A7mtOaC1WvcUU7NXNj9pTxrpXjbx1Z3+nQQwAQBJvJwc88ZPc19A6X8fvAWk/CmLSvKTell5JgCAgttxk/jXzh8dPglcfB3XbO0a+W/t7tN6S7cEEHBBGa9T8N/sfwa58OI9an1nydQmgMwjUAovGQCablSavbToO17annH7NvjrRPAnxCm1PVoVe3MbLEWGfLyev5V0H7VXxV0T4jalo39kRLm1Db5sDJBxxXEfB74Wp8QPiAmhXdytrDEzedJ7KcYHua6r9pL4F6V8J7jTJ9JvmuIbrcrxSMCVI707werj7w7Pm3PY/AX7UPhrwz8OdP017bbPbWwj2qOCcdT+NeBfCX4n2ngv4qXHiSa2EkUrSssfpvbP4V7r8Jf2cvBOufDO11LV5/OvbmEyNIJNvlemBXhPwz+Huj+IfjBP4fv7of2VBPIpkzjcqnjn34pxcG3yw9SYx31Or/aS/aAsfitpthYWVn5S2snmszdQcdK6P4b/taW/g34e2ehfYt9xaQ+Wj4/X3rP/ag+E3g3wJo2m3Xh2VUuWkEckIk3blPf611vwL+FPw41X4XLqGvmGa/uA3mNLJtMYHQKKLxtpC67A7cqu9DwDwP8Vn8MfFOTxZJCGaWZ5CgHA3Guq/aA/aBk+LVjY2It1hht5PNAxyTiuf8AB/g3w/qfxkOkzXeNAW6dfOLcmMHgZr0/9qbwj4C0Hw7psnhlIItQWUK3ktnKd81b2+G7/IpRu9WYvgT9qXVPB3gK30CK23JBEY1kUYK+9fOXiC9bUtSurqQ/PNK0jA9eTmvtD9ne6+HGnfC1TrcFnJqNwX89rmIMx7ADPSvkL4mQ2K+MNWXShtsPtDGFR0C54FYYjmcfgsvzM3bWxyUpXBqtuA4PepGhMecnNQtGXavOMgwOc0h2HpQUO3GeaaI9nQ5qSRysq8GnNtB3dBTPJDck80PHu47U7gSCQN05FJ9oUcU3yVjGQab9nU8nrRcQ7zsMSeBSi6V1wp5qPaGYg8cUJEqdBRcAkuFj4Oc+tIzDGe1OaFWbLdRQUGOnFICITiTpUbXG1wuKsCNV5AqPYpYnvQAjPhc06HEq5xinBQetSR7e36UAEfyuBjIr1b4C/Cf/AIW14yi0j7StnEEaSSZjyFHXHvXl8aruxnmu0+G99ren+IoX0ETPft8qrDnJ9q6qFnNXdgR6L+0J8EbD4W6xax6dqh1CGQbHV/vIfw7V6Rb/ALK3h23+EMOu3+suNVuLcTRqjDYMjIGK8O+JmoeJptSSLxDb3FrPwQs4OSPxresx8SJvh+LhLW8m8PxLtSbacBfbnpXsWpxnpP18zZN7pGt+zV8HNG+JfjLUbTW7rybLT4y+1W2mQ5I4J/Osr46/DXQvBnjyGw0S6Z7CRgpMhzt5xnPcVzvwx0/xbq3iF4vCkFxPfsh8xYSQNvvVPx9Y+IrDxA1tr9vNDqCn7snUn2qLU+Vy5n6GTcn00PpH4pfAj4e+E/hbBJZXJk1nyVfz1lzuO3JyPrXDfsu/DHwb4sj1u+8VTK32UiOK3346j739K5TxH4C+Idn4Ki1LUraZdL2Ar5jc7celZ3wf+HfjDxvcX/8AwjIZVjT9/IW2r9M0ONH3Vd+pVp3RH468M6BZ/FiPTbOZotGmuUVmJzsUtg816t8ffA/w98P+EoI/DyqL6NMmdGzv4714L4i8OaxaeL20m7jZtUMwjUZzuYnAwfrXZfEv4ReMfBOi2t3rcytBIit5YfLKPetY+y952fkCnJaHdfs+6L8PJ/h3qd94hijvdbMjIkUh+6OxFeZaBpPheb40WtrfZj8PNc5kQnouDxn0zXQ/Cb9n/wASfEHwxf67p96ljp8BZCXbG9gOcVxHh/4ear4i+IVv4ZhdRqEsxj80t8vGTnPpU81PkVo+pPvXPSP2kv8AhCpGh/4RizhtVjIAaID5h71vfDnUvh3p/wAF44rnTre716ZX+0SzKCQeenpXmnxn+DepfDG8ihutRj1FOjMjfdPpitbQ/wBny8vvhqvieXWIbUzKWgtM5LD39K0jye00g7CblHdmL8Eb7wxpPxUNxrdvHLpsZdoo5D8uc8DmmftIeItD8TeIhPpFtDbquQPJXb8vaofg38IJfib4un0prtLJLdS8sufQkcVmfGn4cwfDrXms7W++3W/Tcww2RWNV/uX7pUYS5ea+h5abfAznmmeRukGcgVMdzMcjimZfzMAZrw7km14Vmg0/X9PnuU8y3inR5FPdQwJr3z45fFvRPFnhyKx0y2hRVQKvlgcY714h4F0JPEvirTNLllFvHczrG0jHhQTzXsPxx+DvhrwLZxnR9Qa4mUDflsg16+G5vYytG5cYtxbuanwb+Pum/D/4VT6A1un22WSQtJjkgnI5rzLwf8QrbQvipH4mntllgWUuUYZGSOK9B+GPwt8G6t8NZdb1e636g7OEh3Y2gZrjvhf4F0HxF8TDpmp3flaVGWctnlgO1dUo1EoWjuK7stSf45/FxPihfRTJGFCtuG0dOK32/aGZfhrYeHUi8vyIRG20YyR3rnvjf4V8NeH9aRdBci3zgjdnPvXc33hPwFp/wps32JNq80HmSSbuVYjPFVH2qnKyVxL/ABHnnwb+L8vwx1vU71IQxvItgIAJU5J/rWR44+Ik3inxVFq5QmSKVZFB45Bz/Sus+A+i+Fby+1SbxAUkSBR5Mch4Oc81z/iaz8Or8RLZIFWLS2nUSonQLnn8MVH71Uk+jZPKn1Nzx/8AHy98b6XHbTIyqqBAp+mKxfAHxc1DwVoN9pluh8m4ctke46Gu9+NkngmXRILfRLCCCSJAFlT72QO5pfhTq/gvSPhvcre20E+rTO25pMEj2/Kt3KtKoldJpC5Yp7nkmh+Ob7S/GkOvwDddRuSB1JB61a+InxG1Pxpdbr9XQ8Ha9aXgHUtC0n4kfar2FG05XYqjfd9QKufGrxNo/iHWIptPgijVD/yzA+76fpXP7/sW+Zbk8sTPb4r+IF8FwaWscv2GJfLEoB2/nWD4B8Za94Zur7+yElna6XEixqSe/Net+Ivij4W/4Vva6NYWMSy+RtY7Rndjr+dcr8D/AIiaP4HOqteW8ck8y4jdxnApPn9rG8ylyteR5V4u1q+1rUnl1EMJxwVYYIrAGPwrsviNrUHibxFc3sKLGHbJ2jFcobdVXjg14GIbdV3d2XZdCJcZ5pDhe9SLGqjnrTWjHB965WMNxHJPFLHIHPFNZc8AU5AF7YqRGD8S3C+F7NMctcs3HsuP614yVO41638TpD/ZWnLn5TJK2PwUV5Ke+K6YfCYS+Jj7aPzZ0Q8biBX6o6EYoNJsYUK4jgRBg+igV+WVnn7VDj+8P519Q6T8Sta06NFjv3IUYwxzXh5lCU3G3Q9HC9T6+Vge9O46182ad8cNXgA80LL+ldTp3x8R1AubVge+2vC9nI9HQ9qprLmvOrH41aLc4Dy+Uf8AarobLx9o+oY8q8jP/AhSs+qBnQ7RzSGNW6iqsGpwXH+rlVvoasLIDRsSRyWcEindEpz7VSufDmnXQIe1jPvtrTJppoA5iTwBpjuSEKA/wiium/WindgfmMtTRn2qGpYzX2bPORqWL/MDX23+zfefaPBcIzwrECviGzYbq+xf2XrwN4bkj6bXrxMzjelc7sM/ePd93NLTG96TPFfKnpjZX4rB15t1jMP9k1sTNwaw9WbdbSj2NXHck+dvh7N5firxNagYDF8flXznqcLW+oXUJ42Ssv5E19C+Ff8ARvidq8OAPMJJrwzxpamz8TalF/duJD+bE19jlr/eNeSPFxq91M57+KpW9ajYHdUhX5a+jR4rPRvhrMWt7hDxjBFd7FdGPhRXnPw1Y7rhSewNekRsBgcVx1fiOyn8JNu+UHvTY5GZgGFPXBIpxKdA2TWJqTM2I8AZNRfMw5FG7Z1NSqysmRQAyMyBh6V9Afsw+F/C3iDxN/xVBV4VH7uGQ4Vj74rwNJVDAHiu1+Heg614n1uGw0JHlu5TwqHA+prrw9ua0iteh7j+1d4G8H+ErrR7nw0sMMlxuE9vbvldvY9eK9W+CPh/4b33w5hivbWxkumQ/aHuVBfJ9DXzL8Vfhj4y8EfZZ/EsLGGT5Y5g+9c+mRXT/C34H+NPHfh19Q0m6W2tFyFWVyPM47V22g1Zv5l2ly7lPwlZ+F9L+NdzBchZ9Dgu2ECtypXPFex/tP2/gS6+G6XWkpZx6isi+SbcANjPIOK+Z5PBmsWvjhtAmjZNWE/lMued2eufSvVPiV+zX4q8I+DzrNzfx39tAoeWFWOYx3PvU+4+upd3dNnY/so+JPB+k+H7mHV47db9nyZplB49Oa88+Nl94Ym+NEdxpUUJ0sGMzLEAEZgeeKpfA/4I6l8VGuza3w062hHzSHkk/SsT4rfCnUPhn4yi0e8uFu2uAHimX+IE4p+5e/UNea99T6g8dfEnwJq3wivrUQWxla0KRwbBlWxwR+NeH/su+P8AQ/AetXU2qwq0kqBRKwBKeuK6+f8AZDvrjwKuqx66Hu/s/n/ZmX5OmcZryv4G/CVfiZ4wk0y5u/sVvDkyv3OOw96f7q+mxPTc6v8Aay8c+HvG3iDRp9GWOSSJG8+ZR94dgcV6d8O/2kvC+kfDmHSprNUeGAxmJQArcY5ryn9oz4A2fwjjsL2wv3vLa6YxmOX7ykD+Veh/Cf8AZT8MeL/h7BqeoX0zahcpuzCQFj46EHrUKVNLVaD05dWeJfCH4kWXgP4qS67JbKbWSWTah6IrMSMenFek/tMfH3R/iV4XstK0yEkrOsrOeSMCvM/Cfw1tLz4wt4V1C+CWcN00UlwMDKg/4V65+0d+z34U8BeB01vQriSKeN1VopZQ4kB7j3rTmg7XjqRJaplH4H/tMWfw58HrpF3aCQRkkFeMmvIdU+Kb3Hxel8YQW4XdciYJjI4r2j9mn4Q+CPG3hW4u/EEiXN8z4SNpdmxf8a8j8e+B9E8P/GiXQrS5L6N9oRRKWBKqeoz7VTavZR1JSTe56D8YP2rV+IPgG70BNP8ALa4UK8rCvliaMY5GcV9t/Gz4TfDbSfhHPe6QIYdSt4VaGSOTLOeM5HeviK6VmHy1yVmr6RsibWRFtG3Ap8SDj1poG1eeTSRRtvznjNc5KPavgj8aNU+FK3KafH5ouOSuMg1Q+KnxS1X4m+JItTv18uS3ULGmMY5zXX/so33hfS/E1zd+JI4ZESPEXnKGVSe+DVz9qLVvCOseM9OuPDsMcY2YuWgUKrc8HAr0VzcqfL8zZJdyGX9obxt/whX9nKJhYtD5HmiM424xjdXA/Dfx7rfgfVpJtJd/PmHITqa+uPD/AMWPh5p/wlj0xooMrZ7GhMSncxXrn1zXzr+z74s0Dwt8SpNR1SJTapv8kMu7bk8fpWvPPnTUf+CVHlscj8SfHPiDxtrSXGuGUTRDaizAg/lXX6J4x+Id14PddNtr2bS4kKPLEGKgY5rf/aq+I3h3xxqWlPokCK8IPmSqgG4H1r0r4a/tIeF/DPwzt9Mmto1uIIChVQMMcdTVRnWu+Vah7tj5Z8Jaxren6+JNKEjX8zbdsedxJNbPxMk8XTS2p8TW91AeTELgHBrU+GPxF07wv8VpfEFxbI1q0ruiMOFya7H9pP456b8UNP061sLdU+zuXLDk/SpvV5LLYrS5jeAfDPxL8S+E5JdChuJNLjUqDv2ggdQPWvPPDuj67deLFsNPSYaxJKY9q53bs8g/jXvXwr/alh8E+AbXRjaIZIEYA4ryXwj8U30H4pt4rMCszztMUx0Jo5ql1zP0JVtdCb4sfDfxr4Nt7WfxKkptpTiOQvuAPp7Vc+Hvwg8c+PvDEt1opl/s+PIAZ9qse+PWtv48ftCyfFPR7bTlgEccbhzgdDUvwu/aa1D4e+DRotvaq4QnD47mtI+0b+L3gdrao8n0Xwhrlx41TQLWGQauZjD5eeQw613Xxd+BfjD4e6XaajrcyXVrKwQlH3bGPQGuX0n4nX2k/EhvFqLuvTM0pGPWur+Lf7RGsfFHR4dOu4/s8CsHK8DOKzfNZ2loO6vtobfwf/Zm1v4meEp9Vj1VdOt8lIUbJ3kdenSvB/G+gXPhPxFf6TdkNc2sjIxU5Bwetev/AA3+PnijwR4VbStLillt1JJ2pkDPf2rxvxbrdz4i1y81G8bdcXDlmNY1otRTcroiVrHNPIWY8VA0hXAAq3JtOajGxeuK85mBDuIHTJpqs3cYFTuAKbuUjAOaQiBmfdwOKVt2OOtS71U9cUhYDJNAEa7sYam7ZN2QeKm3hulNMwU471NxETbucdaSJHH3u9THHWhZN+QKLgRSRu7Ag4FOZWZcA8+tEk3luFxmlZiq7gKVwIo4mjUgnNNa2ZpNwPFTxSF+oxSPNh8AZFO4DWjLR470sUZQYqRuFyBTodzjkYouA+G3ywYn3r2r9mv4kab8L/G39p6jbrNGYWRdwztJ714zEz+YBj5a9p/Zv+Huh/EXxmbPXrv7LYwxmVhu2l8ds100fi2uNGt+0H8XrX4natb3MUSr5bZG0dvSu0vP2oYLr4Y2mgwwR25itxE6ooGSBiuE/aE8B+GfBviCNPDc8jWjcMrvux7g13d18MPhxa/B6zu/tDXOt3EIkaVJOUYjpivehGTqfw/+ACVla5wPwF+Nn/Cqda1W5EKst8mN7L905zxWF8UviZJ408XQ6vjLRyCRQw44IOK7f9mPwp4I1jXNYk8VMs0don7iKQ4V+eTjvXL/ABe0zwxa/EiBdKjFrpEkyh1U8Bc8kVmueMH7vzDlV73Os+IX7TF94y8Kpp0kexfJClAMAHGK434P/G/UvhnY6nY2qkxXmGO3rxXs3xss/hqvw7trTQbKBLmOEFblQA5OO9cn+zHdeB9P0PWrnxBaQ3eqBtsSzAEBcdRSbq80PcS8hxtLqeMa943vdR8Yx68hIuYpllTPqDkV0nxK+NWreP7SOO93DjB3HiqviKfw/J8WImESx6K1yrSRx9NueRXof7QXiLwZqmi2sGhWNvbeSAFaJQGPHU1UZVbTaaE4x7nC+AfjZrvhHwvcaFaeYbJ2LnZngnrmuRsfG2o6b4ri1y1cpdo+5cdee1e1/BXxp4O8P/DC9truxgn1iRmV5ZlBOO3WvLfCWvaDpvxUi1G/tI5tKWcsYD9z2/WjmqKEFeyG4w7lHx78QNV8XOv28OD1+fvVnS/GXia08KiBYJ5NNjUqsu07VH1ro/j1420bxVewvptvbwhT8vlqBgenFdlZ/HHw/H8JbbQYLOCGWOHZLlQS7Y5Na81V1Ze/Z2FyU3ozxbwP4g13S9amk0RZnu5kKskQOSM+1ZvjnUtV1DUz/aySQ3KfeSTIP613vwP+KGn/AA68TX99NbpIZkKxeYB8n0rmPjB4yh8a69JeIoBJOCoHTNc9Rv6q25j922m556zA8L1pgkEbYI5pRGFXPegRruya8Eg2fDNne6hrFpFpyyPeM48tY+pOa7r4meF/GHh8RnxDBLEkoyu454965T4f+KH8F+KrDV4wGNs+7aa9A+L3xtm+Iy5kyTkcV69BpUZe9byBNLRoy/C/w38Z6t4Rm1PTkdNKTcfmbAb1xWH4F8J6z4o8TfYdMLJfLklskEV2Gh/H6+0vwJH4exiOMEDaOua5PwP8Tp/BfiibV4ky8qsDjrk962vTXJef/AD3b7DPH3hDW/CusLaaq3mzNwrA574xXSeKvgv4l8L+DrXVL24X7PJGHW33nIU+1cp44+Il14w1RLuUkujbgT61qeJvjVrHirR4rG4f5Y0EfXPAGKh1KSlP3xvl6If8LPhPrHxCkvH06ZLWO3U75WbA+lY9x4H1FPGsfh95Y3uZZhGsu7KnPen+C/ijqXgi1urezkIW4+9zWJc+ML+bXo9W8zbco+9W9DWftKXs4+9qJWXQ9A+KnwZu/h7awtJqsV7wN0aAjbnt16Vd8F/AeXXfBM2v3Gppaqc+XEp5P1rgvE3xK1TxWireSlgOOtN074jaxp2knTI7gm15wpJ4rT2lD2j1dh82t7HQfDf4a/8ACbeL5NHmvFto4cmSU+gPaofih8PrXwb4ijsbO++1xyHbubqOa5bS/Fd/oeoNeWtw0UrghjnrVPWPEV3q9wJ7iQtJ61zutRULdbj5tLWPWfGHwX0vw74Kt9QXUzLfOm9kyNv0qL4O/CfQ/FejahqWsXhQQ5WONWwScda8uv8AxtqWq2sdtNcs8SDaMntVfS/FWoaPHJDbXLpFJ95AeDVuvh/aJ20HzeRY8daLDoWtTwWkpmgDHaT1xXNksF5q7qF893I0kjFmPrVJj8teRUcZTbiIYC557UFSx64oRv8A69DSbW44rBjE6HHelVQuMnNIzfxYpqsW56UgOW+LEwW30qNTx5Tvt+p/+tXlXX869K+LTBbyyTutopP4k15uo6V0x+E53uXdD0+XUtWs7SBd000qxovqScCvoS8+FvinRxmbTpWAH3lGa8r+B+n/ANofFfwrBt3BtQiJHsGzX6VtChXBUEe4rxcdVcZqJ6OHjo2fCclrfWfE9vLDjruUimpeOnGSK+3NQ8L6XqSlbixhkz6oK4zWvgb4c1RSY7f7K5/559K8z2ie52nyy+oNtwarSaq8eCkjIe20kV7tqv7NJbcbO/wOyuKwP+GaNZaQA3EJX1ya2U42JdzzrSfiFrmkyIbe9dlzwkhyK9b8F/HSeTZFqltIo6eYg3L+lbPhn9nWx08o+oSfaWHOBwK9L0/wNo9hEEjsYSB6rWM5Ql0GrmJa/FDRJ0B+1gH0IOalk+JWjL0utx9ga6H/AIRPSCc/YIQf9wU5PCulJ0soh/wEVz2j2Gco3xN00sdqyuPUIaK7JfDunjpaxj/gIopaAfmPUyn5RioalU8V9ezzy9at0NfV/wCyveF9OuV6ANXyfbivpj9lm6KzXSZ+XAryswX7hnZh/jPqSYsyHa2D60zzCqAE5PrRngYqB2618gesR3ElYuptuhcZxxWncPwaxdQb922OeO9axJPnSCU2vxguO28da8i+JEPl+NtXDDrOSP0r1PxBN9k+L0R5UMAK88+Lsfl+NrxiMCTawwfavrsD7tRPyPHxnwfM4RutLw1OnQo4DD35pq8819Gjwmdt8OGH2qZQeStekqpkXIODXmHw+kK6ntzwykV6ekvljgZrmrLU7KXwlqMFV60qw7W3d6bG25ckUokbdjHFc5sS+WJODT0j2LgHio2Y7cryadbsxB3DFMCZYQwBzk+lekfB/wAeSfD3xEmoxA7gMcV5tubcNo4r034I6PouveNLKDXif7O3fvEzjd7E100bqWhcT0L46fH5vipodlp6xCIQv5jHBGTVj4R/tIX/AMPtD+weX50K9FI4Feh/tG/C/wCHejfDttT0KK1stSiK+ULeT/WA9QwrL/Zb03wHdaDdf2/b2s9/I33roAgL6DPSu6L3fLp2HJxaPDtc+Il5qXxGn8UKNlzJKJBt7Yr0Txt+01rXirwXcaJLH8k6bGlK9RXN/GDTfDGk/GFrbR9n9jF0Z44z8o9QK+gPGF18Mr/4Q3UcVrZxTx2v7naoEgkxxz9aLu3w/wDALtG6bPnT4V/FbWfh9I40zfIW52KM5/Csv4nfEbVvHniaLU9R3RzwqFRWGMc5r0L9lnxJ4X0DXLuTW4o5JG+WN5lBC/nTv2r9b8K6t4m0mfQUhV9hFw0KgAjPHApc07bfMdoJlK0+O3jWPwjJZxRXD2XlGMyhCVAxjrXm/gjxRq/h/XPP0tpDdyN92PJJOa+tfh18Y/A1j8NotOuYYI2jtyjRlAd5xXzp8IPG2jeE/ixNqtzbJJZ+Y5iVsYUE8EVpzT5tv+CZe7rYofFjxd4r8SfZI/Ekd1CI8mMTqVz9M1oeBfFXxCh0CSLw/BfXNpEMM0KM4A/CvRv2qPjN4b8feF9PstKVXvFm3mTaMoO4q38AP2jNC8B+CY9JvbdQ6EkspAJPvTTqN6LXsLmilc+b7O+1ZvExmhMx1V5uVwd5fPTHrXc/EzTfiDZ6Dbv4ksb6HTnwVabJXPb6VU1D4k2Ufxom8WQRRLbNdmRYgOnueMfrXqfxx/aa034gfDmXRbeBBPMVy4OencDtUqU4xdiuaLZ5P8K/C/jPxWZIfC0UriIZdlO1R+Ncv4q0fW9H8VT2WrxSRassgDI33ix6Yr0/4A/H+P4U2N3bPCrxzNubI5Nch8U/igfHPxGPiNIlTaV2rjjipk5uKvLQfNG5v+K/g/8AEW38D/2vf28j6WkfmMu/5lXHUivCbhxGvNfT3ir9q671z4e3Ggsi5mh8ksBg4xXzDcYZiSO9YYjmuuZ3MpNPYiDBxkChJsNg0fTgUq7fxrkJPSPg98N9T+J3iJNN02QQtjc8rHAUVv8Axo+C+q/CPVrWG/ukvY7pcpMh547GuP8Ahx8R774eai15YOY5GGCVNXfiR8VtT+JF9BcahMZDCPlzXoRt7NPm+RrGaTtY918Ffslv4q+Gy67LrX2W6mhM0cKpleBwGNeR/Cz4Vz/ELx4PD/2hbby2YSzN2AOOPWotN/aA8S6R4dGjwXskdsI/LCq3auS8N+ONR8K6s2o2U7RXDZywPXNX7t173/AB1Euh63+0T8BYfg6umzW2qNfxXRKlZFAZSPp2rsvg7+zToPjT4d/23quqtDcTKxRI3ACY6Zr578afEzWPHjQnU7t5xH90Mc0/Sfitrei6aLCC8kSBV2hdx4pxcbtOXzKVRaM6v4f/AA7s/EXxU/4Ry7vAlnHOyPPkDKg/1r0f9pj4I+FvhtodhfaFds00kojkhaQNketfOFn4mvdN1I38Fw6XLEkyA81a17x5qfiZUW/u5LgR9AxzU3hbcPa6+R9Zfs9/CHwHr3w7/tLxA0c19MWyHk2+WvavFdH8JeHm+NzaLNdbtCS6IEm7gpnpmvL7Txxqum2v2aC8ljh/uK3FUF1WZJ/tAkYS5zvB5qXKDad2T7Rn1l+1B4N8AaH4TtJ/Dggi1FHAIhbO9ff3rQ/Zuh+G9v8AD+W48QQ2txqTlvMFwoY49BXyDeeI7zVAq3NxJKF6bjmo4dcubRWjimdEPVVbAq4uknZt2JlUb2PWdNj8K3XxvZZFVPDRuyfL7bB2+leq/tN3nw5k8I2aeGrK1g1FJAVkt0AO3uDXyWb1lYOrENnOc80TatPdYEsjyAdNxzQqlO2q9Be0Z9l/s1fEnwN4R+HMsWqwwf2jI7eYzRglx2GT2r5V+KV7p2qeONXutKjWGwmnZo41GABXL/bJYxhXYDOcA8VHNMW56msKjhKPurUnmZCIduec1H5AkbnrUis2fmHFMdm3cCuUQ1o+1IsAUZXrT2J2mmxq/elYBPJBPzUrRjp1FDRyM3y9KkKNtwOtQ7gM8lV6UGFOpHNOjjfoTS/ZnZsk8VIupAV3NjtUixqq8VL5JOQOtEVm8fU1NxkTRLuBIyaDCu0mpms2ds077GWTHSi4FUKpxijaobtmrMVj5a460GwLPntSugIFVVyT0p25W5HSpzZllxTo7LYBxS5kMhWQKQO9aek+ILrQbgXNrM0MuMZU4qoLHHzUv2MysBjJ9q0p1XBqUWFuhc1LxJe60we5mZ8cjk02PxXfpa/ZBcN5GMBfSqbWrL8uCKQWDLHu2Nj1xXT9cqczlzu7J5ejJbXW7nS5TJbzPG7cEqcZpl9qlxqD+bNK0jf7Roh02S7bbHG0jdcKM02S1eJijLtYdQRU/WJSXLcOUVtcvbiHyZJ5GiAwFJ7VDb6lcWjEQyNGD1wasT6TdW1us0tvJHE3R2UgGn6fod3qW/7LbSXGzlvLQtis3iG9XIfI9rFCa6dsuSS3WomvJrhQJGZvqauNZytcCARsZSdoQDJz6YqfU/Dt/o2z7bZzWu/7vmoVzS9tra+4uUylupYyQhYA9cGmvI2M4zXQab4K1rVrF7yy025uLVBlpkjJUfjVKw0S81O9WztbaS4umOBGikml7dd9gUW3ZGWJGbrSbm7cCtzXPCupeG7oW+pWclrMRkK4xW3J8KPEtv4bXXJNKmTTmXcJGHUeuKl142u5DcXF2Zw7b+i0yRWx713HgP4Y698RrqeHRbQztCMu3Zaz73wTqen+JF0S4t2jvzJ5QjPc+1T7eF3G+qDldrnJiGTqaFhdm9BXp3jj4KeIfAOnw3mowKIZMcqemfatbwJ+zr4h8c+FZ9ftjHBZR52mU/f+lZfWafLzX0BRbdjx5o2PFHlsq/ezXeeEfhhqfjDxoPDluVjutxDMx4UA9av/ABV+D998MNUitLi5iuxJwHTsfQ0/rEObkvqK2h5msZ5JOKjaIt3r6N1D9liTT/htB4im1aJbyaHzhb9sYziuG+D/AMHZPidrF3bvdLZ21quZHY8n2FZrGU2nJPRFcutrnlXlkZ70kcOwEg5Nej+JPhtFoPj220EXglhlmWPzvQE13Hxg+BOjeB9BhvNO1BpZcAsjEHdQ8XBOK7g426nz80O5smkaPIxXv/wo+Bug+MPAt3rmqam0M6lljhjOMY9a5P4c/DzS/EnxIGi312EsIyxMmcFgO1L63FuSXQm1le55bHCFHFSrCOT3r1r44+A/D3hHW4otCkIhb7yl9w+orsNS+FPgrT/hdbah9qMmqyxCQvv6HHQCk8YlGMrbgrPqfOjQ7veoJISM+le3/Afwp4W1i+1OXxCyyJAv7qNjgH3rlPFOj6EvxCFtaN5WltMA3P3VzzinHFKU3C2waWPNvLCjGMGmCPvXtPxm03whp+n2kWgogmRQGdTyeO9YHh2Twta+Cbg3kSzarJkDd1H0rWnN1I3sF0eZcNmmuvGBxW7oDWEesbrxA1sM4X+VRaz9lk1B2t12Qk8LXoxw/NFSuS5GOsZCdKRYWOSBW3etB9mVUwD2qG22NGBuCYOSD3r36eTUZYj2MqySte/6HLKvJQ5lEyfLLsBjml8oxyYZcH3q4rRi4OeFPG4dqddLGsaoDvbP3qw/svDrBzrOr76e3kN1pe0UeXTueX/FicS64VGCI4I0yP8APvXABeldj8Q5N2u3wzzvx+grkPSvEcbWNFuepfs1xq3xk0Bj0jZpPxCmv0Gjv1ZRzX5ieFdXudC1iC+tJ2triLO2RDgivbPD/wAefEtqFB1ITj0nUGvm8fB8/Mj6TL8HLEU24s+1luA3epA26vnDw/8AtE3bALfWUUo/vQvg/ka77R/jrod3tW4Wa0J7suR+YryOZbHoTyzFQ15b+h6ltFIVFYul+MNH1ZQbXUbeUnoA4zWwrLIAVYMPY1aZ506U6btKNhCKVeKXnpQBTZkKtO6U39KWkA7IopMGimB+XIHc09aYOh7U9OlfXHnlu34wT0r6A/ZkvGj1yaMH5WHPNfP0DYzmvZf2dr3yPFir/wA9BivPxkXKjI6aLtNH2nE26NT7VXuG706CTMCn2qG5b5a+NaPXvqU5peozWNeycMM5q9cyYYnNY+oOGzzg1UEB88fEaT7J8TNPkx95hz+NcJ8XFZvFsrHo0aEH8K7X4yMbfxbpk44xIP51xnxacya5bt2e3B3fia+swXxRfkeTi17jODbv3+tA9KRh2Apyg19FE8FnTeBQ39pKy9Fr1WBl28nmvI/B8xh1WPkjcccGvVol85QeRUVlqjpovQ0VwQCMVKpU+1VYVKr1qdYyxzXNynQP4HOacrBvu0nl7hQLfy2yDTUWBNvCd6u2OoS2EqywyFGHTb1qiFB5PJp4U/QVpG8dUBv6l4z1DWrZILm6eRFP3WNVbXxBd6bn7PO8Wf7prLMYXBpVG7/69bKdRu9xFqbUJbuZppZGeVjksxyaWTxBeSR/ZnmcxHtmqm3b7UNGMD1rZRqO6QuaxJHfyW2WjdlPbBpJNQlvG3Suzt0yxqLZQ0W2p9nUcW7OwuZbXHHU5oV2K7bem3NNWdvvA4NN8vd0FAUetDp1ElJp67C5ltccbiSRjuyfrVeW4kB+U1O2Ap5+vtUO0DkkKPU9K3hhcRKp7NQfN6EupBR5r6CCZmHJOaas0m7J6etOkj2nBIBFMDIxKB1LqMlQeQK0jg8RUUuSDfLvpsQ60I2u9yQyOfumnpI6pg81H5iRJvkJVAcfj6VKrJMoaNt4Pel9QxXsfrHs3ybXtoHtqfP7Pm97sQM0m/OeKZMpbgVJLPDHP5TP8/pj9M+tI22ONndgirySa65ZNjoyhGVJ3nt5mSxVGSk1JWjv5EUaFepzTfKYtnOBU6lZow8bBl6H1B9DUMl9DFcLCc5PDN2U9hWtDIMfiK08PCm+eCu12IqY2hThGpKWktmSrluKeMqMZzQzCON5WHyICTjrTLe6W6Vio2FTgqSMjPSlDJcbPDSxcYe5F2bG8XRjVVFy95i+XuJJ4p7fMuO3So5LyOGQxttwACxZtp5/u+pp8h8mN5MbtoLbR3rpnw7j6aoylGyq7GMcww9R1IxesNxvl7elDQ7uSKis7w3TOn7t2TBLQklcHtyOtJqN5NatHGgUBl3DKFt5z90Y6V30+FMdLGvAyspJXvfQ53m2HWHWJV3F6bD8HpR5e3PHNSsGjV2VN7quQv4VR0u5u79ZWuI9gQ4HGM+1Yw4cxE8NWxPMrU3Z6/kdEsdTjUhTs/eLKx7jnGaNuKS5aWGNfLRm3d0wctngH2qyI280AqA+PunpnHT86dTh2pRw9DESqK1T8PUmnmEZ1alJRd4+W/oQtGFxxg0JGWBOM1Dax3ZklacMI+D+8xnd3x7UXkNzIo8lWdcEDY+3a3Yn1FekuFoxx/1KdeNrXv8AoczzJ/VvrEab326kvln7uOaHhPpjt0qUK/3TIPMIx5gHAOKp6bp91arMZ5PlbGBvLZbPLc9Pwrlp5DRnh69aVdKVN6LubSxk41KcFB2l17E/2c7dx6etNWAs2AKranpdxfzRvEY2AUoRKeEz/GPetD7OTGYfPYNs2edjBzjrTrZLgaWHw9b6wm5v3l2FDF1p1KkHTso7PuVzGrKdsivg4+U5waayKiqWZU3HaNxxk061sZLeRnleNflCBIVwDjufeo73SP7QkRjMUQDaV68Z7eld39hZV/aHsJYj91a9/PsYRxmLlhvaOlad9rki2zSPsUZbOKkuLCW1UNImAe/WpoWW3mBGSo9TzTpnUW5jVmYs24lua8KWAy+OFrTdR88X7q7o9H2tXnilHR7mhoPgfV/EdpNc2Fr50MX32ziq+i+HbvXNag0u3i3Xk0nlqp45ra8O+Orrw7pTWcIOxmJ+U461naH4iudH8QQ6rBxcRv5gr5etClGEeR69Tti3fVG78QPhDrXw5a2OqLHsnHytG2cH0NeofD/9kHVvG3gWPxCdVtbJJkMkccmSdo9fSvL/ABt8SNU8dOgvn+WPkJXW+Hfjf4p0rwmulWyyPaRoY9yg4A/CvErucV7rN7q+iOD0fwPPqXjmDw2sifaZbr7MJM/KDnGfpXtfxt/ZUh+FPhKHV4NcF9MMCWFkCn6r7V4dpGtXtl4lh1SFi16k3mq3ctmvTfih498Z+INHiTXtPubS2lwVkniYB/oSKznKpdL8TOTaeiOv/Z5/Zo0T4neELvXdY1OS2RZGijjiIGMdzXnNv8KbNvjFF4Ra/X7E9z5f2rgfJ1/Or3wpb4gzaTfQ+FbS6u7MHdN5Y+VT+PeuRjj1+68ZKiRz/wBumbCxgfvN/pis3Kpd6jje57B+0X8CfC3w60ezutBvmkmGFlR33bvcV03wV+A3gTxB8K/7b1y682/uFYgLJtMWOnFeS/Ffw74/0WxtH8VQSx28gzGWORn396tfDvwL8RNc8I3V1oSSDSEBb5mwD64rLmlGlrItqbewnwt+Hvh3VfjK2iatdZ0mF3O7ON4HQV0v7T/gbwZ4YurJ/DCLAfuyIrZB968q8G+E/EPinxxHpelrIutNIQecFMdSa3vjF8NfFXgC/t4/Ek32lphlZFfcv0pu7qK8iPeue/8Ah3wj8Kl+CtvI9vb3OpTWuZJ2P7xZMc89q8q/Zp0nwdJ8QNTPiJY5ra3QtbJNyrHPf14qPSf2b/GNx8Nz4kjvUgs2i85LUykF1+lcj8H/AITa18UPE0un6dOLJ7dd01w5xsHT8az5laT5i/eOk/aRHhWTxtbS6FaxWsJI81IAACM9eK9c8eap8Novg3FY2GnWZmNspDhQG346g+ua+e/i18LdQ+G3iaPT7u8jv2lOFmQ9TnHNd94r/ZnuPDvw1g12XxDHLdvEJTY54xjOBStFxheZEpNdS3+yl4p8GeGRrT65aW82pNjyWuAD8mOgzXn/AMV9e8Pap8Uo7+ztIl0/zlaeOMYVgDzW58APgLH8WIdRvb3VV0ywsztJUjezYz37Vxnjb4croPxCTw9a363Uc0yxx3De5xzWvPH2so3YtdHc9h+OHxP8Ja94Ft9N0yytlCxqsawqNy8dSaofs9fFbw14H8F6lY3dpbjU5WZjNKoJKnoBWR8W/gFo/gXwlb39jrhvr5APPj42g9wKq/Bn4K6N418JX2t6vqbQFXZIYYyM5HrXNFUVQ1btc1jzTejOP0/xppunfGCPxEbOOWyjuN/lbfl+uK679oL4u6d8SIYY7WKPKtuDKoG32rjfD/gGy1j4nQ+HZb4Q2TTFWuM/wj+tdF8evhj4d8ByWzaFePOrHa4c5zjvXb+7VSK1uc0l72rOh8E/tEWvh74Wx+HTaxwywoUJVBl/cmvPvhf8SrfwN48k1ya1WaOQMPug7Ce4FeieE/hP4LufhTFq1/eGXVrmIv8AK2PL9BXCfBfwj4e8ReOp7XXZ/wDiX26M6rnHmkHpWMfZctRpPzNI/u5XUiv8YfiUvxC1qK6hj2iPkFgOa6fV/wBoa71rwPBozQ7DHCIT/d4GK5340eH/AA9oviZY9D/d27feQHIXmu/8aeH/AADpvwwtU0+CN9R8kObrd85YiqtSUIe6Eved2zgfhH8Zb34XxanHbwGVbzkleCO2K5/WviBd6n4wh18oPPilEiqfY9K9S/ZwvPBWn6NrNx4it4Lq+3bYVmx8q464rgdRm8Ot8VIJFiVdGa6DSIv3dv8AhVRlD2s/c1/MzbVr3LPxF+NGqePrCOC7QxIOMGp/Cfxm8SeHvBp0W0geSyj3fvFU4ANdX8fvEfg7VNLtYNDtbeJ48BWhxn8TW18Pfix4U0b4QJplzZW/2/awkYoN7GudteyVqfUcHF9dDwzwx4y1fR/Fy6tpwaS+YkFFBJbPan+PvGOteKdS3asjxSr/AAOCCK634M+ONB8K/EifU9Qto5LSRW8pXUFUbNRfHDx3pfjHxIlzYQRqEbJKKBkeldKcvbpcmltx2hbzE1fxd42k8D2q3SztpWzYsu04xjAya5PwFqniTT9QuDoCTSyOv71YgTx6mvXfGX7QGl6z8O7XQ7W2VGWFYym0DBxzXH/BT4uWPw5TVEuLZXkuuUkxkj2qU6ipyfs9RK19Tz/Wb7VrrXDJdCRdQEgIBB3Bs8VseMpvFr6fC2uJcLbt9xpAQDTNc8dLfeNk1xYVIScS7D0IB6V1HxW+NiePtOgtYoBEiAAjbW3v80PdK925z3hXw741uvDtxc6PBcnS+SzKcL+FZfg3wn4h8VeJkstHSQ6pkncG2lfUk12/hH4+XHhnwH/wjyQZ2hgJOxzXLeA/ideeCPFkutwDMkisrDOOtL9777UV5B7tloVPHnhPX/DGsCz1xWNy3Rt24H8a29Y+F/izS/CMOqXIK2LLuWJnOQv0rI8e/ES68cawNQm4ZTlfatjxF8btV8R+G4tLlOEjUJlfQU0q/LDReYNxvsM+GXwn1zx5a3t1p062sMPDyMcZPpXNTeEdQPiv+xDh7xpfLDbuD71peC/izqvgnTrqys5MQ3Dbj7Guem8WXv8Abg1ZZSt0r7leq5avPJvboRHzRreP/h7qHgmWJbu4WYsOinOKq23w7vLnws2tmaNIeqxk/MaoeIvGF/4nlEl9M0rD1qoviS+GmixFw32ZT9zNdNPn5ffeppKz+FDNB8Py63fNDHIse0FizdKZeaW9reNAWDkHGe1R2l5JZyF43KMe69aJ7hpG3s2TXq06tJRV1qZWlfyH6lpxtI0bduzUENr5kYJzkjr2pJbmSdV3HOPWmC6aOMoDgV7VPH4ONeU5UrxtZLszCVOo4pJ6jFT94cDft6jPWnTQBWjJGN3bPSq/2loX3KfmqS1YzXce4/xDrWE8dhpYP2Cp+/e/N5dg9nJVOfm07HjfjmQya3fN6zGucUc8VveLT5mqXTZ/5at/OsRV9q8abKRveEtNOpXzxhVbamTurt4fB8zcx228/wDTNsGsL4bR5vLpvRAP1Nel2kzQNkHFeDiJe+e/hJShD3Wcr/ZNxp7YaO6tz/toSPzrRsbq6Vgqzo/+yWwf1r0XRvEAAEc6hx/tDIrpE8P+HdfjxPYQkn+JRtP6V5cuR/Ej6GhmOJo7SueZ22pXNvtaSBgRzuXP9K6nQ/iBqNg6tb6lcw/7Jfcv5GtuT4J2c37zSNVutOfsN25RWVffDnxlpPC2un+I4R/eQJJ+YxXJLDxlrCVj24Z7CS5cRSTOy0340azGBve3uv8Arom0/pXT6d8alkIF3phA7tbyhv0IH868PN1YaXI0ev8Ah/V9EPQyW7eZGPfDAfoTW/pOmeH9aVV0nxhZrK33YNQVoX+nNYezxEHo7m6rZJiV+8hys9ytfiloc4HmPPbn/ppH/ga1rbxlol1jZqUAz2dtp/WvFj8N/FNunmRQQ30WOGgmDZ/lVObQdfsz/pGk3Cgf3Y8/yzWXtq8W+aI/7Hymsr0q1vmfQsep2kq7o7mJ19VcGivAbP7csO02VypB6BWFFdsZOUU2jxamU0IzcVV2PkLinrjjmjZxSqvevsmfCkseVPvXqHwLulg8aWhzzmvMY145rtvhZcm18YafIBn94oP0zXNiI3pSRrT+JM+9LWTdbIfao7iTrUGnTbrGIg8Yps8nWvifI9kzryQ81i3c2VNaF9N1x1rCupvlI961ghs8J+O37vUNPmXoJB0rk/ieTcLpM2BgwkfyrrPj0f8ARbdyeFb+tcZ4yZbrRNJlAGVXG4emBX02E+weZivhkcVIu1uBQKfIvTPWmqvXFfRxPn2avhkf8TS3PYOM+3NetNcR2NqjH5md/LTsCexPtXkugzeRfRn+8QD+dev29vHdWvkSjftPTHf1r6PJ44N42k8dHmh1RjW9q6E/YO0uhZsphdbl+USIcMFOQKhvNYNndSxhUKQAGTc2GYH+6O/WrNnbwWcaRwABepbPU1PNa2s0iTTxxs4OFZuua+wwqyaGaVo+wcqdvdVtvkefOONlhYL2iU1u+jHTTeVavMqltqlto6n2qlp2pG6kMbSRyErvxFklOcbTWrxuIxwahhFmjyrbiJJM/OI8Z/GscEsvWCxNOWHcpN6Psv0N68cQ69KUato9V3KupXz2SwhF+8ctIykhfbirtrM11DHIY2idhkqf6Uk1xFaxl5mVF9WqeOdJNsiOJEPKsDkH3BrHFfVv7LoxjhbNPWXcdONT61N+1umvh7GNa6jPcX3lyCUSb2Uw+WAqKPundV/UDMlkzQjc+cHb94Dvipn1S2+2GEt++ztJ24Gf7ufWia4S1jMrkqBgdM9eAK9nEVYvMMLUhguXRe618RxU6dsPVg6999exBp8kzwKZwyk9N3UjtmqNwLr7e/7uQlmXyXD4jC8ZyPzrQguhdbvleN1OGRxhhxxUc+oeVIYxE8ixrukZcYVScZ61pg6mIhmmIUMIuaS+HsY140ZYKk513ypr3r7kt1HLJbyxwnEpXCn3qvp8M8e/zI/LiwoVWbcc9zmrLTLGpkz8oBJI9Kp2OtR6k7oqshC7hu7j1rlw88VLKsTRp4dcier7HVWhR+t0qkqj5ui7kl9azSTLJEizR7NgRmK7W/vVYVSrfew23G4euOar3F8YJ44VCl3Vm+ZtowMd/Wn2sovI4pEJCyAEZ61ni3mFTL8JKdJKKa5X37XFQWFji6yjN8zWq7ehVstOltZCzhV2qQTH/wAtDngnPcVLqGnnULVFWRUdDnDDKn8KVblpGi5UiUNtRclkx3am6hcvY2wkUDc7BAzZ2r7nFexWnmv9s0ZqMVVa07W8zgovASwFRRbdNN33uSw2osxHDvM3lgLuPeo/7LjivpbnzmcNkhT1yfU0tpcG9hjmZdm/jjOODjIz271Wa8mOrC2VCVLbfLC5O3H3931rlwdLNfrGNhTcVKz5v+AdFeeE9lQlNNrTlLc1sLqJUDNE6uHV0PIYVYtIFtVIDs7MxdmY9WNVbsyx2crwKTIpGNvYdzSaVNPJahp0KuScMwwSB0OK4f8Ab5ZE5Oa9mpbdTpf1dY9e6+e2/QsS6TbzXf2kls5DlM/KWHelaNJFaORcqwwaWVpvOCRxsRkbfQjHzFv89hStC7bxER5hUhd3r2rTGfX1Vwftq6u7Wf8AKZ0J4dxrqFJq2+m/oRw2sdsgjj6ZySeSaSSxtpLpZ3izKvIx0OKbYwz29viX5QWyqk5Ycf41X1CxuJruKWFl2HbhtxHl468d810UY1f7WxFKeLSbjrLv5ESqR+p05xoNq/w9i+pDOwKblbgj1qNI4YY9kMaoAcnHc06SEziSFJTEWUqHHY+tQWNi1halJJVeRm3BVyQo7/414uH5aeW13LFbS+Hv5nfUbeJglT6b9vIs/um8tnRWZeFZvX+tNVv3m4NncefQ1FNp4uGikEqptGGDDJXBzuX0qYMjySB/usGBPrmtcV9X9jhZ/Wr33X8pnRlUc6sXStbZ9xiyxbWWHYQDg7CDzSPfR2gUSyLGT93dVay0tdPWQiTcZMYAGMAe3rS3OmpeeW3nvC4Uq20/eU9q7Jyy/wDtbkqYpuFvi63tsY3xP1TmjSXN2LUbbpFIbJPIOaT7VFMGWOQOFODimxxxxbUAzGo2ge2MVFb2FtY7vKLO7Db8x6AHoK8ai8t+r4nnrPmT91d/U65/WPaU7QVuvkSy3iWkQZ93zc4UZOO5pyyKxVgwIbkMOnNRXENteRos4bKHhlODjuKl3RxuPLUbFHC+wpV8Tlf1LDqMm5p+8vLyHTjifbVOZLl6f8EhW/S8Z1VjlecN3HqKjur8WW0bS7nJwCBhR1PNSxLbW4kEEQQuck9TjPSmyfZ5oQk8ayBW3Lu7V1PG5JHMVO0nRt87mfssbLDtXSqX+ViWFhNtdGyrAMD6jrUFrqSah5qIrI0R5DDqP6VJ9pbeGBxjpgdMdKVbiIKwjjVGblmUYzXnUcwyyNLERqQd38Gu3qbzp4hypuMlZble+1A2OwBBucFgzk4ODjAx3q4zFfm2Mfl3hcfN0ziozcqFUMoYA5G4UfaCZBIvWlXzbLJYehCNH34v3n3Jp4fExq1JSqe69l2IbW6a63sRwoBzgjGf4Tn0pLy4uIlj8lfvc9M7iDjb7VO14GjK7QoznApi3nkgAeufpXas/wAthmH1iGH/AHdrcvn3OZ4HESw3snVfNfcmjXN0isu0E8rnP4VYvUj8sOsflMGI4PUVnecfvAnOc0v2lpjhhXz081oujWpexV5vR9j1I0ZJxfNtv5no/gSbQLfQrr+0bZLieQn5nP3R7Vz/AIZm0y18Z2cl7H5mmLPl0Pde39K5fzmRvlyKVpDgeteBXxEalOMFG1jrjzLqevfHDxB4X1o2H9gWcFs0eQ7QrjI9K9T+FPxi8IeGPhVFp1xp1pJfCNhK8sYZyTmvkwTM2S1M8193GRXg18Mqqs2a88tzvPDHiLT9P+Jlnqk0CHT470TmMgY27sgYr6C/aN/aA0zx54RXTbNY2UldqjBK++a+RfMKrx1pYbiVhl80pULtO+xErs+mv2fv2jLP4Z+EL/Rp4gJnlaVZiOSCP6VwVt8WFsfi9D4vSNXEc+/aR1GMGvImeTeMHipWYtHgHml9XjdsEne59B/HX9ohPidpMdkikoCGHGMVF8Pf2mJ/B/w8Xw7tKiMMq7f4gfWvn0ZVcE5qFt7SZzwKn6rTceVj1PTPBPxbuvBnxCfxLCuZJCwcD0arXxc+NN38TriIzEhI+eTyTXlEhLDjrQvyIB3rT2MObmsK3U9vsv2ltah8EReH2ZisMXlKwPG3GK47wH8WNV+H+s3V/ZSkG6XbKucZrz/b82c0silqn6vT1VtxnZeOPiNf+ONTW8u3IKfdGelX9U+M2ua9oqadcTZRUEZZTjIAxXn+OMdaasaoxPerVKCSVtiXFPc7Hwl8S9W8DrcJYXTJHcD5488E+tZGseKLzV9V/tKaZvtO4OHB6Ec5rDaMOQc0u35cdsVapwvzW1HY6TVvH2q67C0NzdvJG2MqT3qvo/jjU/D9rNaWl1JFFKclVPHvxWEIwowOKb5YYk96fJG3LbQErbF1dYube9W9WZluFbcJAec1PqniW+15hJeXDTH/AGjmsllz1PFOCjAx9KvlW4cqvc0bfxRe29p9jS4kEH/PPPAqtHqU1jMJoJGik/vKcGqu0UrBcZNHKgsWbjUJrxjNPI0jHuaZ/atxJGI2kcoOACeBUO0NUjWciRCTYwTsSK1jTclotiG0tGMW8ljb5Cw+hpHmbbnndmmH5WOeKlhha4kCKCzMeBTjTcnZLUbaSuyNppJD8xJ+pqNZJOg+7WhdadPaMomiaMsMjcOtTw+H7ySza6SEtCoyT7V1RwdaTcVF6GPtqaSbkrMymkbgCk3N3NaGn6PPqcrRwIXKjJ9qg1Gwl064aGZcSL+VRLDVYwVVx93uUqtNz9mnr2KI3d+lNbJbjpT927INN3bWxjJrksaiNnbj1oVCo5OTSs3GcU6P58GqUbuwtiLadxOaGXdiusPgtv7JF01zGkjLvWM9TXLS7o22gV24jB1sMouorX1Rz0cRSr8ypu9tGMAxxSrHtGRU9nam6uo4t2CxArode8NQaTapJFMZH6OpH8q0o4CtXoyrQWkdxVMRTp1I05PVnK/ZyzZNRtCTwa7XR/DdrdaSZ5pSsjZ27TwMVm6PpEWoagYZHwigk474rqeU106af29jFY6k+e32dzmmhxwKFhA+tdD4k0mDTbzZbuWTHfsayEjI5NeXXoSw9R0pbo7adRVYKcepEtuGbOKVo/UcV3HhbTNPm06SS5UNJ93r0rmNXt0hupkiJ2Bjtrsr4CdChCu2rSOelio1asqSTTiZbR9gKhkhO0krVxY9pBP1rrbxtKOhbVVDJsyG75xRhMH9ZhOXMlyodbEexcUo3ueetGPSrFggN1EcZ5BpJowrHHQGpbHKysQM7I2b8lJrzOtjrex4Nrz+bfSt6ux/WqEa96takd85PTnNRRrWr2MInd/DWE4vXxkfKv8AM138cfpXHfDOH/iXXsnrMF/JR/jXcxr8o9K8Cu7zZ7dDSCJIQVxWxpupyWrDBOKzI1/CrEK+tcjOxSPRdC8SbgoL4NdxpusLIo5xXh1rM0LghiK6/Q9dZWCk1hKJommevoYryLy5Y0mjPVZFBBrA1j4P+EPEW43Gjwwyt/y1tsxtn14/wp2j6iJlXnFdRaykgYORWN3HYTPMv+FC6jobl/C3iy+089RFOxZfpx/hUq6v8WfBe1buwtfE9qOskWFfH4Y/lXrcDFsVoRE7R3rRVG9zNnjH/DRAtP3WoeENShuh95VjyPzIor2prWGT5niRj6soNFa80SD8uFXNOjwOMU0cDGKevavqTyh8fWuo8ETm38QWTrwRIDn8a5pFzW34fbydTtXzwHBNZVNYtFw3PvTQpPO0mBh3Ufyp91JhTWX4PvvO8P2rL/zzX+VWrqXg18O48smj2ltqZOoTfgK5+6mPzYNauoSHJrn7yQ5NbwRR5L8c18zRR3wRz3riNeG3wrpPO7eBn24P/wBau5+Mi+b4flP93mvN/tT3PhKzRmDLGcD25r6HCbRfmeXiXo0YUnOCOBTF6HipG+bnGMUgWvpInz7LOm/Ldxk9jmvZbVGuLHETBJHXIY/TpXjFrxMtevaRMXsbdx3Qfyr1MHjXl9eniIq7i9mRKkq9OVJ6XNHT4ZoFPmBEjJ+WNe3B5/lTtS0uTUGg8pgAvBDfz+tEczOeRirCyOvAOK+ljxZWp5lLMYU0nJWscn9lUpYVYWTbS89S7DH5caRtIXZQBu7k461lafobWF48rOCArbdowWJOefpV1XZuc0nmSbh6V5+H4oxeFhXpwStV3Ompl1GrKnOW8Nhuo2AvreMLJ5U0bbkbGRn6U7T7NdPhSPeXbJZie5PXHpT2yenFNXK8MawnxJjZYJYBv3E7+ZosDRVd4hL3mrEP9iQLeNcCV/K3+YYc8b/WrVxBDfQvFOu6NuTj+dQ/M3IOKcc7etPEcT5jiKtKtKfvU9hU8vw9KM4RjpLcbBaw2a4Rmkdj8zsck46U26sbW8lSWTIYAD5TjI9DRtKkjPNNZSx69KxXEmZRxUsYqnvy0ZX1HDOkqDguVdCzuRWOFG08Fe2CKrWtvbWZfyVCM3U+3pRztxTEh2tnNclPOsbTpToxqPlnuu5rLDUpSjNx1WxJcRQXSqs8ayBemeoqTzlXAUbQBgY7VH5PmZpPJ2LzUTzXF1aUaEptxjsuxcaNOMnNLVj1uEVmKqqu/DECj7SI1I4KnqD0NQNF+NPEe/g0SzPF1KirSqNyWzuSqNOKcVHQd9p3ew6ADt7UG8/hOd2MZ74pGh2sRimNEvXqaxeOxEZSkpu738yvZxdtNiT7QY/mHemi8aTnoOgo2q30pqr6DisFiqqhyc2hXLHexOt4ynAJoaY7aYqrT1jMnAFRKvUna7bsHKkMFy0nXimNM/QdKszWb2+N6Mg7ZGM1X3Be+KmUpxl72jGuWSutgEjbc96VZGPUU+KMyMAoySav3WkXNnGsk0DRq3AYjg1rCnUqRc4ptIiVSEZKLerMty/AA4oO4L05rc0jw3fa3u+xwNLt64FU5rGSG8Nq6bZ1bYUPXNa/V66jGTi7PYy9vSlJwUldbmYPM3c9Kc25uhxXQat4N1TR9PS8uINtuxADZ5H4VgSPs/Os61Grh5ctVWZVGtTrx5qUk15CDKjrSeWw5qWH98wVRyTgV0upeDbjT9Hi1AyRPGwBKq3zDPrWtDCV8RCVSnG6juRWxNGhKMKkrOWxyrQmTAziniM9M10/hHwoPEU0oa4WBEHVjgVR1XRTpesNZiRZBvAEi9CDXR/Z+I9lGty6S2OVZlhpV5YZS9+Ku15GL5JWo2jDda77XPBtlY6KbiG9EtyBlowO3euCkVui9anG4GtgJqFZWbKwOPoZhTdTDu6TsCx9BTvs+OgqxYwiWeNG43EA/nXb+IvDuiQaL5lpNKbyMAnKgKfUVrhMtrYujOtDaJGKzCjha1OjO957HBCLJ5pwg+Yha7PwbaaObWafVI3mYnCKhx2rKurW1OueXErJas4+UnJAPUZrplktWNCnXclabMqeaUp4iphlF3grvTT5GDJb7e2Ki2qvJ/Cu98VNpE2lH7LbeVcpgmQnOeORXBupcg5rizLASy6r7KTvp0OjL8bHMKXtoRaXnoCx56VLJbtHgMpH4VZ0dUXULczYMW4bs+ldZ4w1Kx1DTYYoLeOJ1f5SnUirw+AVfCzxHOly9Ca+MlRxFOgoNqXXojjYrOSRSURnwOdozTUhMsoRVy5OAveuz8La9Ho+m+UscTO7ZZnHI44rHtbyGDxALsKuzzN444BrpqZVQhToz9qvf38jnp5hXnUrQdFpQ2f8xj32nz2ODPC8Wem4Yz9KoeYqnGa7fxp4mGt2scbEMwbI2jAWuGeJd5J5NeTmeFpYSu6dGfMu525fiK2JoRqV4ckn0HKw571IsgbgVCO4pyqE6V5J6Q9pljbBHNJJJsXcKVlU9etBUNkHpUgRLNuBPSl847tuKTaval+VfrSASR9oyOaRJS65IobA96TI6igBqyszkYwKWSRlA20qsvTvStIkfJNAChjtyeuKYrMc55p/mKVOKYkqscZoAa27cMdKcWOOvNI0wQ470rONpIFADY93O78KaVbzOKWOTdjjFHmHdjFMBXUlcdKI12Yz83tSSMVXpk0qtuXkYoAjWI7sk05oy2McUqsSxGMCtvQNHTUnkaV9kUfJ9TXbhcNPF1VSp7swr1oUKbqTeiMy1jVZELjcoYE11ut61a3WlmCPBUqPLwMYxWHfaWLbUltkfMbEYYjsa2PEPh+10/T2MEzPKg53EYbp0r6bC08ThqNenFLTR/8AAPHxEsPWq0Zybu9v+Ccc0e5smtLQblbDVIZ3GVXrxWftbNa/h+1iu74JOfkx69favGwHO8VT9nvfqeniuVUZc+1i3r+tDUFRAPuHhj/IVPH4qRbGOFVKlE2lT0zVXxNZ20TI9uojLD5lB/WsFVO6vbxuY4rB4uquZXejsebh8HhsRh6fu+6tVc2dJ1Y6c0xX/loOoOCDVHWtSbUplLAAgdq3vCX2FYp3ugrzAjYr9hWL4jWBr6RrfaFY9F6VliZ1/wCzIXmnFvbqOlGl9em1BqVt+5j7QORSbV3A05Y9vJpPL3NnOK+TPbHRwGZtqKWPYAVK9q0DbXRkbrhhWn4avoNPvhJKuRggEjOKteI9Rh1GWN1GWUcnGM8V7VHCUpYX27qe9fY4J16ka/s1D3bbkLapetpqoQxgVdobHb6/jWBMw8zJ612t54gt7jSfICbR5e0Lt4zXGPGGbJFbZolFwiqnPp9xlgrtSbhy6/ePt1ZplMQO/PG3rmtLVft8ar9rVgCMjPequnzCzuoZQu4o2ceta/iDW/7VRUAwinI5owzgsJN87Uu3curz+3glFOPcwrfVJYVMCMwQnO2tLQ9LutWuCLXKsoyXzjFZqQ7TnFa2hatLpckojGVcc1ng6nPWjHESfKaYiLjSl7FLmKetWc9jePFcncwP3s5zVKFN30rR1a8bUJvMaqkeVFediuT28uR6XOijzezXPubul6JPcWbyxyKg7KT1xXOXnmLIwPXOK27XUrqKz2ICYxnnHTNZ32eS+uMRo0j+ijJr18VGnOhShRT5ranLRc41JyqWsUolLMq4wWOOa2dR8Ny2WlJP5ysSMmP0GKyri1eOYxOpV/7pGDV6+nvvsa/aN/l4wpYYH+elZ4SNOFKr7WLv+RVZycocklb8zmZd28mp7ZittqDdStnP1/65sKZI2WPemXjCPQdYkyVIs5Bn6jH9a8Dqd3Q8FulJkBpY096dcDMnHNSRpkDitZGMT0v4b25TQpWPVrhv0CiuwRT6Vz3w/i2+G4mPO6Rz+uK6lI6+dqv32z3KS9xCIv8A9arCrTVj9asRx5xWBsEan0ya6HRLNtynHNZthaNLIABXeaDpfC8VnN6DRr6LC6bf6V2VgG2jms7TNPCgV0Vra47cVzF3LNuDxxWnbrkCoLa3Iwa1IYcYyKaJbEWPiirax8UUyT8ptgpyjninKlO24Ir688oeuK0tNk8u5hI4O4VnheQKu2p2SR4/vVlPaxUdz7S+Ht2ZvDFoSefKXP5Ctq6k+UnNcZ8J7vzPCtuP9iuoupODXxtWNqjPbWyMy+YNnmuavpMSGty8bg1zmoP3rWGw2eefFZfN8P3Axng15ZpCrceGdrDJRiR9etepfEQmTRLoZ/hryTw82/SJo84Oea+gwvwHmYgrs7SLsPCA5FIq1Ky89OKQfer34ngMIR++Ue9eu+G0L2FsoyzbBXksY/fIe+a9j8A3QtY7WZgOORmumnTVapGDdrkubpxlKKu0bcmlz28atJC6g9MjFS2emzXzEQxs7eijNdjrXjKPUtBNm6IXHAYAZ+uayfDevHRSxjbY394jrX0NTKcLSxkMPKr7rW54eHzTG1sDOu6FqibtHuY0tpJayGKVGjcdVYc1pr4S1N9PN6tnI1uOTIBxS+INa/tnUFuSecAE+uK2IfHU8OnmDzW8srjy8nFY08BgXWqwqVbKO3mdFbF49UaU6VG8pW5lfY4psxtyMfhUe4NnFSyMXYknNM2gdBivlZJJtI+jV2rsYZNuB2pwbOKTAanLhSPSpQMj3FmNDNt6VKSvbik4HJ6UgIxzzQrHceKk2g854pgYdO9AzrvBPh/T9UuC2pXLQQA4+UcmqnjbRLTRdUWOwma4tnTcC3Uc96paOL64mEFkrSO/8KjNM1qC9tbrbqCNHNjKq3HFfVyeGlli5ab509z5qNPExzRzdZeza+D9THbczdOK6XwXa2M+oZ1FDLbqMlFOCa51pArEE1q6BY3WqXsdvZqWmfgKDivJy6UY4mDnHmV9j1cdFzw04qfLpv2Nrx/a6Rb3UL6RG0UTLhlkOea42RWY9enauo8U+F9S8OtGL1PlbOx8giuZkk2t0rfOOV4uTjDkXY5sqVsJBe09p/e7jMEjAojUx8E5p275ScUiOWzxivDPXAoS2c4rX0G5itNQglmUMqNnB5H41ks7DGOavaXbm8uo4j8qscE114VyVaDiru5jWUXTkpbWOu8YeJYfEGlxKyxLLG3BRcHFcLJGG69K7PxH4ZttN00XFvdeY+4Apj9a4yQsDgV7efe2+tXrRSdlseRk31f6qlhpNxu9y7pMwtbyN2GQhzXTeIPFsur6WbWSQsuRsUr0565rmNLhFzeRxucITya63xBpejx6KXtDILpD1OMGvRyv608urqlbl69zkzCeEhjqHtk+fpbb5lLw94qk0O1aKF2iJbLEHrWXcas9zrX29vmO7J6c1mMG7Uc968ipmuInShRb0hsenTy7DU6068Y+9Pc6PWPF0mp2L25LEN/e7VzBUckml8s7s02SMtxXFjcbWx1RVKzuzowuDo4Kn7OhGyJLWb7PMjp1Uhq17zxJLdWskBOEfqM8fhWEI9vfmlWHB3ZpUMbXw9OVKnKyluaVMPSrTjOcbuOxuaLdXUavHbBmB5KqKq3t1PJfGWYlZgQfTGK2fCOuJo0c3ADt3xniszxJqQ1XUXnVcFgMkDGTXv1m45bTmq12n8PY8ilG+OmnRsrfF3I7zXpbuMxlsbvvY71mNIFXmm+X1PSm7QevSvm6+Jq4mXNVldns0qNOjHlpxsie3kLSKUzuzxj1rpdS0PV7XTRc3UL/AGcgEt6Vzlr+4njkHO05xXVap4tn1CxeJi5Vhjax4Fe5lvsnhqvPUcX08zy8Y8RGvS9jBON9W+nocwt5JbkhHZcnoKktg95cRquWkdsA+9RCAydAT9Kms2azuEkX7ynIzXBSnNzgqjfLc9CSVpcq96x0fiLwjd6TpyTyXEUycFlVuRmuOdjkAV0Wo6rcTWrI6sN/dq59iFxnrXZnnsPrK9he1upw5X9Y9h/tLXNfoW9NtzeXkUOQpdgMmt7xF4dj0u2WWO6W4bOGVewrn7FnW4R4/vg5HFa+pfbFhUXEbRoxz0wDWmBdD6jVU4Ny6MMSq7xVNwmlHqurMdmdfu1b0O2jvNUginYpEzfNj0q7pvhu/wBWt3ltLdpUXqQKzre3n/tBYkVhcBtoXHOa86jTnCrTlVi3Fv7zqq1IzhOMJ2kl9xveN9K0u2tI5tNeQsG2srd/pXBzB92RxXW+JtF1PSY4nvVYRydPauTkkAfHWnnTg8U3ThyrsY5VGUcLFSqc/mC5ZSM80kSup+al3DrTo5N2eteAeuK0TSODnApZIyy4FDSFWAxT3b5SR+VJgQLCUHJzR5PzZJpY2Z+SMUm593tUgOkj3LjNNEeFwKWRm24HWkXdjJoAQQgNuzQ0QkOTTVDFie1LIhOADQA7yxSLCE9qcFbb15pscLKSSaAE285NO2jHJ4pHhLNnPFOaM7SooAVdrYwKT5dx9aI49nfNN8n58nrQArBduTSBg3NKybsCl8vAxVAMWQbj3q3bahJZE7GK7hg471VWIKc1Js3deta06kqUlODsyZRU1yyVy4skl9dBsl3c8etaes6XqNjarJc5MfT73Ssqxna1uI5VGdhzitbVNZN5blSpG85Jb09K+iwvsamFqzrTfP08zy63to16caUVydfI55mG6tDR9Pl1C6WOLAYAktntVIqN1aGjXUljdboxksMEV5+B9n9Ygqu1+m514jn9lL2e/mO1zT5tOkAlYSAjgg9Kx1kJY8c1sazdTXcmJl2leAoFZigbvaqzD2f1mXsr8vmThed0o+038jZ0HQxqUckjS+UqnH1rM1izaxuniJ3behrT0ma8jikW3VmB5wozWVqM0klw7TZ3k85rtxXsFgqajFqXV9Dlo+2+szcpJx6LsUBnHIpuW3cDipVYH6U3eBxXzh6omT2qeHJkUOeMjNQEhamhVpGVQM7jgVtSb5lZET2bOp1XSLKHSRJE580AHdng1yDKScA10OoaJd2disshyvdd3IrAb5WA5r180f71Jw5XZHDgvgfv82pd0WOGTUIEuP8AVk/N71ueJPsEflmCNCxGPlJwPf61i6TYnULoRKwXuTirWuaW+lyKDIJMjrjFb0OeOAlL2aavuRU5ZYqK59UtjWjk03+wceWpm28+u6oPDN1a2kkzzqC2Bt3DNc5DI2MHpXReH9Hi1GN5ZJNgHAFb4XEVMViYezgrpGdajCjRlzydmzM1to57x3iG1W5wKoRx4q7qlr9muJEVtwB4NV41IxurwMRf20uZa3PSpW9mrbHSadrEFvo4ttgDbSCSP1rO0XV00e7lfbvVxgH0rWXRbP8AsfzWkzKy5GG6H0NcpMm1yM9K9vFYnEYb2Tla6Wh59GlRrKpFX1epb1TUBqF/9oC4OeOPSrWt6wb6zETIAODn04xiqFjCjXEYlOELYJzitXxJb2EdnH9kCq5GCdxOaVGtWq4erU5lruaTp041KcOW9tvI4uZQvvVHX38nwjrTDqYlT83Aq/JF+8OTWb4xYw+DL7H8bxr+ua+Y6npvY8Um+/Vi3Xio2XLn1q3FH2qpvQxij1zwbbiHw3YKB1Ut+bE10CKOBis7w1AYtDsVPURLWyq+1fOVHeTPeh8KEjjq1DFuYDHNJHHWvpdmZphxn3qDQ1fD+mmRlJH6V6No2mBVX5azvD+jgBTj9K7mxsdqqAK5Ju7Gx1nZhccVs21r04p9pZnjIrWt7XGKkRFDa+1WlixUyR7aftqxXIfLNFWNlFOwrn5QbKcq5xgVY+zn8KlSEr0HNfVnmFcxleoqeP5cEckGphCSuSKctuV/nUlpH0z8GroyeHYsnJ9K7m6fK+9eZ/BCctou30NejXHK18nWj+9l6nrwd4oy7x+DXOam4VT3roLpHlyERmP+yM1k3uh6hcIdlnKfTIxRHco818afvNMuBkH5TXkfhaUwpcABSQx4YZFe+ax4E1TUIXRligDDH7yT/DNcdp/wRn0mO6ln1JHJBYRxRnH5mvZw9SMVZs4a0W9jziZRnjr3pijFWbqBkkYEchiKh2EV9BF6Hz8twhX94vGea9V8Fwm4t4I2fGTjPpXlsa/vB9a9O8GM8NuCp5PFdtFxU4uWxGrukes3nhPTk0NrmG/JuFAPlsvWuNZmwcVseXqL2JfyJDDj7+04xVGzsp9QuFhgjMsjHAUCvoM5hSrVaf1Wm1dfeeVlXtqNOp9aqqVm9eyKi5brTWVyeCQK1tR0O70eQJeQPCW6bhWhpHg3U9btnntLffGvU5AP5V4EcDiJzdNQfMuh68sbh4U1WlNcr63OZ5701VZW5NWrqBraRo5F2upwQfWqyyBmwa86UXCTjLc7ItSV47CtGWHDYpyxkqK6Lwz4Tl18vtuIrdV6GU9faq2raJJperPZM6M4P3kOQa9H+z8RGjGu4+7LY85Zhhp15YaM/firtGKIyrUskO7g9K7S68DRW+im8TUY5JQuWhA7+grN8N6PbavcFbqbyIxwW711yyXFxrQoOPvS2OKGdYOpQqYiErxhuc8Idqj0qMwgNkV0PivSIdFvxHbT/aIGXKt3rA+bk15WIw9TC1ZUam6PUw+Ip4qlGtSd4s3vC+uPoVwZEbaSMbgORUnjHxE/iS4gkkO8xrt3EVV8NWtrcXga93CAdh/FWh4utdMi8mTTA4Q/eDGvrIUsXUydyTXIn8z5upUwkc4inF+1a36WOUaMda1dC1JtLuvNj4OMD1rMZCc1q+HVt11CE3KCSIclG6N7V8/l/N9ZhyOzufQYxQeHmpq6tsXte8QSavGiSFjsP8R5rnJFAzkV3Hje80y8tIJLSyjtZlO0iPjI9/WuGljL9678/jUji37WSk/I87I5UpYKPsabhHsxvFKduOOtJ5fy4oWAg5r5pXPeA471ZtZDHIjL97PFVmi3Edqu2Ti2uI5CNwU5xXTRX7xa2In8L0NXVItRjtVe5ikjikxgsMA1gthetdfrfiybVNL+zzMXUY2rjpXI9eTzXt5vGEay5KnPpueZl05ypN1KfJrsSW+5pVCZ3E8YrpNX8M6xY6Yt3cQuLU4O7Oetc5bzG3uI5E5KnOK3tQ8UzX1iYGLFW42seK3y5UHhavtJtPpYyxjxPt6XsYJx633XoR+HPCt14kkkW32KIxljI20CqOoaTcabqElpPHtlVsex96m0vXH01WVTtDdcVDe6s95eC4b7w9azlHL1hYNN+0vqKm8f9aqKSXs7e73ubeq+AbrTdH/tAzxSKAC8asCwrj5WK9BzW9d+JprizeEnhxg8msFm65rizL6opx+qbW1OjAfW/Zv63bmv07DAxPtShm3Y7UuRTlcdP1ryD1DpvBukWOpTSNfzNFAg/h6k1U8T6da6bqjxWkjSQdVZutHh+xvr+Ro7NWdschRUGt21zY3jxXilZl7NX1Nb2TyyFqbUr7ng0+f+0JXq3Vvh7eZjsrb/AGprAtxmnNICx4pN23tXy57xYtMeciu2F3DNd3rT6M2hvDDabZlA2zbvmJrg7ceZIq9ycCuv1TwbcWOhreNcwtxuMIf5q+rymU1hq3LS5lbfsfOZkqP1ig6lZwd9F3HeCdUt9Jjmklt4rhydo80ZArM1i6trnxAZo0WOFmyyIMAVgNcOnKdams91zMiE4LHFT/aftaNLCxpq8Xv3NVl8adepinN+8tr6HaeItfgvdHFqIYUCj5WQDJPrXBtHubmuw1zw7aWemieG882dQN0e3j65rjpN27ijiCVZ4he2iouy2Mcjp4enh3HDybjd7l7SpUtryKR+UVs10niDxJ/aliIGcNhsgAYwK5rSYlnvIkkOEJ5rqPE1npcOmq9mriVDySeDXVlscR/Z1Zwty9e5OOeHWPoqonz62tt8xuh+LJNIsPIido+edvesRdYFvrn24rn592K2vC0OmtYvJfReazHAAbGK5bVIUhvZkiOYwx2/Sox1TEU8LQqOSstisNSw08TXhGDu9+zNTxN4m/tq3EeWb5s7mOfwrmNqnrTtmDnNM8ks2c18rjMVVxdT2tV3Z7eFwtLB0lSoq0UG3npmnpt7U3Z2p0UPl85zXAdY/KKcE80pwFJPQUySMFg1I/zLg9KQCCQMeOlM8wBtuaFULwBSCNd2epp2AkZtvzU3cGGcUuB3pdo4A4osBGsgZsCh5PLxxmpRGqk0hC0ACvlc4pqu7dVwKl/h9qarDsc/SiwETO27AFOZm29OaeXVWxnmkJCjJPFKwDIy3ekO/wAzOOKkWQSdKXzBuK96LAMk3Mvy9aVVYLzSySeXQsm5c1QDI1YnmneWzYINJHJuzgYp/mFWAAzQBd0uOE3kXnnEe7n0rf8AEk1lJZqkKIrKflK1iaTZnULyOHd5YY8t6Vp69pMNjEGgm83B2kEY/GvqsFGt/Z9WUYJx79TxcQ6TxdNSk1Louhznl/Pk1u+G7mG0uGaXGdvyswyKwVLFznpW94d0+K+kk80nao7HmuDK+f65D2W9+p0Y3k+ry9ptboN1+6ivptyj5um71rH8kKucVtatYxW9+kcTHy2I69qvarp1lDZYjQpIo5b1r08RgK+Kq1qrsnHc5aWJpYeFKmr2lsV9B1xNNtZEZcPnOaw9XuFvbuSUDG6ul0Gxs5LF5JlWVicFWOAK5nUoUhupVi4jB4pZh7ZYGjGUk4/iLC+yliqjjFqXXsUPLHbikCAZzzUm0gdaasfUk18oe2N21PC5jmR06g5FNSPc3XFXNPaO3uo3kG5QcnvWtC3tY3dtdyanwvS5pahq893a7ZVI99uM1z8mN3vXV+INYtr21CIq57bRjFcoVBY+levmsv39vac9lucOB/hfBy+RZ02aeC6Vrfd5nsM1Pqk1zNc4ud3mejCpPD+qJpV75rpuyMU7XNUGpXwnVduPU0qco/VOV1Ouw3zvEfBpbcrTaXcW0KTSQskbdGNaei2N5cRu1up2dznFJf8AiI6haiHACgYNM0vxI2m2bwKuR1FddOOEoYhNVHy238zGcsRUpfB719vIz74ukzB/vA4NRwnLd8npRdXX2iRpD1JzTFm24I7V4FSUZVb30uelFPl1Ohm0W6t9N85mULgHbnmuXmkxJgCtq48TS3FqIcAKBg4rDmcNIe1duPnh5OPsG7W6mGGjWSftu/QtWEEl5cxxRnDMcCrevaVcaWQsrhs8jFZlvcG3kWRCQy8g1LfapNfczMXPuc1lTqUY0JRfxGko1XUTT90ymBZiay/H7CLwWy95Lhc/QCtncCcDiud+Jsm3w1aJzlrjn8BXn7s2lseUMu6TI9auwx9Mc1WVd0g4rThjG5fqOlTIzgex6TH5em2q/wDTJf5VqwxiqOn7Vt4V9EA/Sta3VWr56W7Pfjsie3t9zAY5rvPC+hltpK5rB8Pab9rmXjvXr2g6OI41+XBrnnLoi9i1pemiNRxg101jY9Kdp9j04rct7MKoOKyt3EQ29rtxxVtYxUqx07b14qrE3IvLNOCmpAtOC1QiLFFS7R6UUAfmjpfgHXNY2/Y9HvbkN0ZIGx+eK7HTf2dfG+pKu3Rntwf+e7KtfeCW6QjEcSoO21QKeAa9GWMl0MVSR8faP+yH4kugpvr60s1xyoJc/pXV6f8AsgafDj7brU0vqIowBX0uqmonh3dBXM69R7stQijy7wt8G9E8G2/lWpuJhnOZXzW+2iWUf3bZM+4zXUTQ8dKzriHb2rllq7mqOcmtY41OyNV+grG1GPKk11N1F1PasK+h68cVmUcNqUZGeOKwLuPdHIPUGur1aMjPFc3OvWuiL2JZ8zaguy7nQ9RIw/WqZX0Fa3iK3MGtXox92ZgfzrN219dTfuo+dmtWMVfmzXoXgufy4UJ4IOa4DbjFdj4PkG2QdxiuqNTk95ERjd2PbbTxzJFpT2m7MbLt21gaTqUmn3ZnjbZJnqKw4mbjFWex5r6Crn1erUp1LJOGx5VHJcNh41YRTtUd2dHr3iOfWoYo5pGfyzkZp+m+Jp9Ptwkcki4HRTXMxsd3zHiuv8KtpEKFr+Dzi3TccYr08sx2LzDHupGSjJo8/MMHgsDl6pypuUIvZanO3lwb64eZ+CxyfrVT5dxxxWnrwt/7SmFr8sOflB9KxmQq3WvjsdGcMRNTd3c+owrhKjBxVlY2LHWJLGFkUkHOQAcVDdak95ceezEyetZvLDGacB8vBrZ5liZUY0HL3VsZrB0I1XXUVzPdms+vzNC0JfKsMGoLXUGtGLK5UEVnbPmzSt83BpTzLFVJxqSm7rYI4OhCLhGCSe6Lt1qD3zAs24rVTzRnBNMCbelP2qcEjJrhqVZ1pudR3bOmNONOKjBWSL+nrNcTxw26s0jHCqK0vEGg6npKxNexsqP909qzdLvm0+6EqHDdMjqK0Na16bVYUR3ZlXoWr6zCRoSy2fPN83Y8HEPFRx0PZwTp21fUn0XwTfa9aNPb7VQd3YDNZcenzf2l9jxtmDbT2xzV3T9duLS3WOMtheMAms37dILxrgEhyTmliKGCo0aM4N3e5VGeNnWqqpbl+z/wTc8TeEbrRLGK4e4hnVuCEbOD6Vy0jleg5rTvtakvIvLZ8jOazfMA5zjNeVms8NOvfDNuNup2YCOIjRtibc3kTWcLXM6RgYLGuw17wjYadoouLe9865UAtGRxz6Vx0M2yRSDg1uXUeof2eJJYpBA/O4jj867Mtjh5Yer7SDcraeRy41Yj29J0qnLG+q7nPMp3EAZNT2qB5kVjtycZq9p+lXGp3Hk28TSvjJx2qK80+WxuGiljaOVTypHNeXDD1I2qOPu3PSlVjrBS1On1vRdKj0VHtJGNyv3yx4/CuIlVugrprjw3qq6T9saNvIIz17euK5mZiuPWvUziKVSDUOXT7zz8tfuSXtOezYtvH8y7z35rsdTbSP7E2JbBZ1UYmDcn8K5C1JlYDuTj6V1F94NnttNa7FxCyqAWj3DdXZlCqfV63JT5tN+xjmHsva0faVOXXRdzlWTdmmiPAxXQeG9BTWrl0kmWCNRne2fy4qLXtFGj6gIUmWeNuVZf5V5UstxHsFibe69DrWPw7xDwql76VzE8nad3WkZd2K7W58I2K6KZlvT9qVN23bwT6VxMqsuQvFc+OwFfAuKrL4lcvB46hjlJ0XfldmMIHanxqMe9MGR1oXOc15h6J0fh3XpdFEnlFlZ/4lqpr+qNqt15rks2Mbm61e8Ii0+1M93H5yKOFzjNReL47P8AtDfZL5aMuSmelfX1fb/2RG81yX26nzlOOH/tOTVN89vi6GBhaRcd6j2/NnNLt3V8kfRlq3fy5A69Qcity81a8e2O5CFYYLEVg222ORcn5c812uoeK45tFazCRCLaANqjdn619XlOuHrL2vJpt3PBzD+NSfsefXft5nFSMOp/lTrdj5qleCDxiopAGb1qW0ZYZkJGQGBxXzlNr2iu7ansyvyvQ2dQh1FbMSzQulu38W3ANYZbDV1er+KmvrBrdnLJ0C+lckcbuelexnXsvbL2VTn0ODL3VdJ+0hyu+xPaq0kyKn3icCtvXfDupaXZpcXJDRP6NnH1FYNtdfZp0kAztOa1dS8TNf2hiP8AEcnnP4UsHUwywlVVZtT6LuZ4pYr6zSdGKcPtN7/IyRdSQqVViAfQ1JY2raheRwg4ZzjJqtuA61JZ3ZtbhZUOCteRSqRc4qq/dTPTnF8r9nubXiLwt/YsEcq3EdwjcHyznBrmmZlYAVrahrT30YTouc4rLaQDI61rmTw0q7eFXumODVaNFLEO8hrNtXpUccjluac0g55pgkD15R2iSGRn46U5s7eOtI0wjYDFK77VzjNIBFDY+am+U5fOce1PWXzKBISxAHFACNGx6GnxxlRyaVmIHSlVmK5IpgCw4JJOaa0QY9acrPkntTJGb+HrSAk2cYp0cKrnBpu47femxs45JpgSNAGO7rTGjHSgs56HinSKXUUAIqKox0pFRc8daVYyoIzmmhfmyTigBJMZGaBj8KZMu4jnFLt+XFArkile1G4KwyeahVQnejgsCadguXre7e1kEiHaynipbzVWulAz09BWf25PFLxjrxXTCvVhTdOMvdfQylThKXO1qg84M2KsQ372rZXg+tVcAGnZHeohKcJc0dGNpNWZamvWm+cnJPfNI2oyTLgscD1NVWkGKZ5y7evNU6tTV3eocsdNNib7dIhKqTg9cUySUnmo1uAM4I+lMkuFFZOcpKzZVluSbs9aTcfTimfaFYZpgulbIzUDJizbhtp0kjfjVT7UFbFEl0uM96QyxubbyeKaNxPtVcXe4dKat183I4p6iLMmTjBpdxUYNVHugrADmmteeoosO5cQEDOabnnJNVPtBxk8Uizlm9qNRlzlu9O6DAOap+cegNSrMdp+YD6mkBY4A96iJG7mq6zNzzmm+YS2SaAuWN2O9KzbvaqTzHtQJWHU0hlraF6c1y/xRcHR9MjzyZGauktwZK5P4qblm0yLt5bN/wCPU47mctjz+GP94K29PiBnjB6bhn86z7VPmxj36VsWcYEqHHfNZ1HoKmj0axvA2Oa6HT281lVepNcTpshbbXpnw80OTVr9HKkxr7V41RJK57MWel+BfD5jiR2GSa9U0nT/AJRxVDw7oYggjULXaWtkIlBxXBbW5pcba2YjAOKtKlSKv4U7bQIYFoxmpMUBaAGbaUDFP20m2mAlFOooAqbeaXZUm2l2k07AR7fajaM0/bShaLAV5IQe1Z11b9eK2tntVa6gyp70hnK3cPBrDvo8g11d9bnB44rAvYcZ4qSjiNZi+UmuSuBtY13WsW+VauJvl2s1XAD538aoV8UaigzjzSx/HmsAr6V1vxCi2eK704xuIP8A46K5dlNfVUXeCZ8/V+NkTda6rwXhpJRg54NcsygtXTeDZAly4zj5Mj866+mplF6nqWk+G7/VLdpLa3aRF6sKhWwl+0CAqRJu24960tD8US2FqyxuQo7KcdutUmvmlvHuWOGZt1fS1sLl0KNGcKl2/iPJo1swnWrRqQSivhff1LmqeFL7SrNbmZFERxypzS+H/DN34kdxBJGnl/8APR8U2+8QTXlmLZnYp05PFU9O1SXTT+5Zk56g4rq/4S6GPi4Tfs7a+pwf8K9TL5qSj7a+nYfrmlzaJfNa3BBlUA/KQetZDSEvjHy1evr576YyytvdupJqkxXsa+VxsqUsRN0fhvofSYRVlQiq/wAdtbAzEYxSqzYyRSb+/akEit0rhOsRmfdx0pxY7eOtM89VyD1pd4HIoAF3Z5p53HpxUazBqc1yIyOM0AaejrH9sjE3zJnJX1rpPE15pt1pqC2tFt5oz1U9a4lZujDrS/bWfg5x719HhM0hhsHPDSp3cup4+JwDr4qniFUa5ei2Z3PhDWrLSYCZbSKdm+8ZBn8K5rxBNDPq1xJboscTHKonQewrLa8kjUhD1qPzS455pYrNI4jCU8MoJOPUyw2WrD4ypi1Nvn6X0+Q7btOc5pcZxntUIlbfz0pXdsAL1zXz9z2y/ausUqM38JBrr7/xm11pbWZffEVwq46VwSyMAN3WnLIyscnIr1sHmVXB0504JWkedicDSxU4VKm8XdHS6Pr0mkiQxsY2YfeWodU1p9Q1A3Ltufjk96wmkJGBSBmUda0eaV5YdYf7KdxrA0VXeIt7zVjqLjxZLJatFk4ZcYzxXOyNuOSarc8HPHpTmbcMGssZj6+OcXWd7aIvDYOjg01RVru5YjkCOCpxg1qTa47W5jLckcnNYSkrTlw2PWow+Nr4WMoUpWUty6uHpVpKVSN2tjStdRey3bW4brSXOpteMrs33eFHpWecE4PSm8DpS+uV/Zqlze72H9Xpc/tOVc3c05NcnkUoW3cYJqg8ncmo8BcevrSMc1lWr1cRb2sr2Lp04Uk+RWFWQNTfO+YrTSwHA60m8d8Cublb2NSyt09vgqSKS4ulmdmXcB23Hmq3mBmxQ7Be9aXny2ewvdvfqHmH0oMhxxQ27yt+07fXFJDmZtqjcfSn7Od0rbhzLuTRyE9aNz9zxUMkhikKMNrDg5FSuHSLdsO3rmqjSqO9k9CeaOjbFZixAFAYjrzTIFa4YhBuIGabI5ikKONrDqKfsanIqltGPnjfluSbm7mmuxb2qS4ilghWRlG0+/SmWsbXhfaQAvrV/VKzmqfLqR7WCV76Cbtoxmo9u3nNNdisuzPOcVZvrU2kaP5quGHT0pRw1Wak0ttxupFNJvcjRQ7ZY8UuOwp+nwpdxyM0oj29vWq6yDzgjN3wTVfVKkVFtaPYXtY3a7E20KBg03YDzS6ksVvtMUjMO4NS2Ets1mxkBMmTnBxito5fUnVdK6TXmT7eKjz2K7qORmouF+lQy3SmbGfkz9OKn1S9tFijWFVDf7P9a56eF54Sk5JcvTuXKryyStuJ8nelJB4zxSaTqtpHbyLLErSE/eYZwPaq1tqMcd6JNoMStkA9q0lg4xjCXOve/AhVm3JcuxdC47YqSNeCQv1OKg1PXEuNpXk+oFPt/ECx6eIeh5/GtPqdGNWUHU0S3F7afImo6jtwZgAOT0okfy+Dx9ao2erG3uN5pL3VhdS7wPasvq1L2fNza9ivaS5rW0NFlkWMOUYK3RiOKS3jkuXKxruI5qjPrzTW/l/1qO11SW33bD97rWjw+FjUSUm49SfaVHF6al6RyjlCPmz0qSaGW3VWcYB6c1iSalI8xfPOc1JNq8twoTOR1qXTwyTte4+abaublvC9xCZAVA9zTbcNNJ5YIB96ybe/njhZVPHpUC300b7lyDVunhvcsn5k81TXX0N68U2rYLBs9xTvs2LQSlxnrgVzz31xctlt3FONzP5e0M2KHHDKcmouz2C87LXU1rdVuZDufaB+dQzSCKbZvzzislWuASVzTWWd/mIOfrSvR9koqHvX3FeXM3fQ37qOOKNSj5NNs3heN2kbDdqxCtw3DZ9qctrcBeAcZ7Vq6lHm5lTIXMlZyNSKeNpgrH5c9qdqE0UD7U4buvpWULC4J+7nFPOn3B6qW/Cs3UhycqgV9q/MbLXVq1iuCDJjnmq1jeRRs5lGR2zVJ7Blj3dOxB4xVRliUkPcxLj+9Ko/rWksU3NTUVoRypJxuXrq7RpG2cAmq/2kIOCaqrcWaEH7XA3/AG2X/Gk/tCyLEC5gPb/WCvMmuaVzdTSVix9oGSTQ90HxkdKqNqWnrwbuFT/vUyTWdLjPF3ET7ZOf0rPlK50W2ufSmiX86z5PEulRqSJgT6KpNR/8JZpyrgFmz1xExosHOjUW4IbNDSM3OOPpWQvjCyHARz6FozUR8bQ5OIpD/wBsx/jSsLnRuCR+lO3OvG04rnh40Vm4ikHp8i//ABRobx4q9LaYn/eQD+Rp2Hzo6DnjAOaRVk3Z28Vy83j2Tcwit3K5+80oB/ILUEnja/bISAf7zOTT5Q5zsfLlb+E09beXrtxXDN4w1Ngcxx47j5v8arN4i1Jn3EAHPACmlZBznon2aQLk8Uv2dmIy6/mK85/trUpC3GTjJ+U0z+0NVkICZBzx8lFl3DnPS1tirYMijn+8KfNDDFy9xEo93Ary5rzVnbPmMM/7I/wqMf2k3/LaTP4CjlQ+dnp/mWS/evIVP/XQUPd6ZCxDXkeR1wCf5CvLpoL8jd58mCOQ0nSq6216f+XtlHTmY4/nUWiPmZ65DrmkW7A/amI/2YXI/PFHifR9C8baLPeW2rR2erabAZfstwNouIgedp/vDPSvJBYzMTuuVcnt5hapm8PvJHsOxkcf3hiloLmbCwjDS5Uhh6itm0hzISawLC2bwndJHO5nsrjjf/zwbt+BrrbWH5gOuelc1Q6KaOi8OafJfXEUMa5LnFfVnwy8Ef2fp8IKfNgcnrXmvwH+Hb6nOuoTITGOFBFfUul6UljCqqOcV4daXNLlR6sNrhY6etuo45q8Fp6x0/bisChgWlxTttLikAzb7UAU/FGKAI9vNLTttG2gBuPain4ooAhZaTpTqTZ8wNaWATbRjFO20YPSkAnWnFQeMUqpS7T3pAZd9a/Ka5q+tz83H0ruJoQy9KwNSszzxSaGed6tb5VgBXAavGI5GHevVNXtflY4xxXnHiaERsSAO/Sl1LWx8/fEyHb4ocj7rRq3P5f0rjXHzV3nxLQf8JBESuQ1v/7M1cTIuD1r6fDv92jway99lVl5ra8LsFvh/ukVkstaGgfLqEPfk/yNdpgtzvo2IUYq4kx2gHmstZto6VMtznrWTZ0l1s7gc59aHYsDziqLXR3Y7U77WVFIC2vFNZByc5NU0vPXio5Lxt4weKVgL5wetHyqx7VQa6O3rzUS3TjO78aLAah28nvS7hWU1w7dGwKVrtuMnmnYVzS3LkkUb171lLOVzljmkaYycZx+NOwXNUSr64FL56jABGayRcFeM5pvmbWznB60WEazXChc9PrTfPXjByDWYJhJ940qyDseKdhGibpc4z70NcLGoOfwrPZwv1pvnbuTyOlFhmiLkcUovEwc9aymmNIZsjAHNFgNRrwKM0Jeq+c5yOlZSzHnPNI0jDpmtEibmkbxt2O1WbVvtUgRWwWNYvnOF6GiC5kjbcoORW9HkU05q6Im3Z8u5uXDfZ5ijHNWGgU2YlEq5xnFYckkszZ2MfemtNOq7QjY+ld8Z4eMpvkunt5HO+dpe9qa9m8c1xtlfC4z1xmkvpIreYiMsV7ZrKhS6wGEJOenBzQ9rdyNkwsp+lYqpD2Dp8mt9yr+/wA3N8jdEls1irMx8zGev9Kp2V3C1wPOPyj9aoLpt7INqxNj1qW38O6nJyls5/lXTKvFyhJQS5fxMtEpJy3LGoXcXmgxfL7VYW9tvsJUqCwX73vVA+G9QwS8YjPfc1R/2DMq5ku7ZB3XzRVxxPLUnOMF7wvd5VFy2JrHUI4ZC7cjGKbf6hHPMCnHGDj61X/su3B+bULdPXLikaz05fvatbAHuXqFiJ+w9hZWvcLw5+e+porra/YxED0XBFU7HVjZzFhz2NRNDo8ec61AT/dU5IqE3OgRn59RkY99kLH+Qq5YyrOcJt6x2ISpxTj3LF/qxuJ/M9ambXjNb+ScgAY4NZ7X2g7gEN9Mf+uDAfqBUiX+lNhU07UnOOpQf41ksVUTlJS+LcfuWStsTWurPayEgjkYqObUDNNu6seRSRXlmx50S+YY6uNufxzVmO+EPMOgtjPHmTjPT/dOPzrJ1m4KHNp2K5o35uXUZJq0kkIjYE4qK31J4TlSRT/td4SDHoFrEWOPmYk/jjFH2jVx0sbKNsZClH/qaUq8pSUnPUFJJW5dCFrx5JNw3dadLeTOuCrEDvVrZrjMu9bSP5cn92DgH8aI21ZpNj3NrHnu0IxWftUr+8VzPsVIbidc7I2z9DTSt0zZWOQkc8CtCOTVl5GqWsfbiID+dHnaiwXfr8mCefJij6d+o61LrLa4cz7FD7PfSYBhlA9SpxViHT7xuRG3vmo5YZZG/fa/eMc/wvtHt93FQywwRqwbWblywGS9y/HPbLVHtE3e4+aXYstpN47HETe/FDeGdROSYWC+oBNU0/s2O4Ehvmkw2R5jh+Pz5qu1vo/yq0m89Mt0H6Gs/aRHzTL3/CP3AJLywxgdfMlVP5mj+xo0HzajZqM/8/MZ/k1Z6nRUwRlu+3ZkUSS6RtTbGd4yGJT37c1PtF2FeZfktdPg+WTVrbd/skn+QpDJpMSEnUkYjrtRv8KoSahpysPLhZl9SoFEmtW8eRHDhGOSv8v50/aJB7xeF1pGTm7dhgH5Yyaka60dcbLmbnuYGrJ/4SCHaw+zAntk00a6i5byVPPr9RUushqMu5pSahosO05vJm7iOE/1xQdb0xQR9h1KQ56eUg/9nrJOvt8wEcYDDBFN/t+QgjamM5xz1qfaFcsjRbXLfIC6PdfV5UX/ABpP7acgsujKAD1N2Tn/AMh1nN4ilznEW7125P8AOoJNeuG4MiY74T9eaXtWLkZttrlzGq7dKt9pGQRcMf8A2QUn9tXrbdun2q7TzuMhz/48KwX1uZufOx+AFI2uyuMNO5GMfep+2YezNxtY1Pywy21mgYZ+VGOOfcmo31nWWUfNbqB12wLWN/bWFKmRmHbnpVeTVlHVyfXmj2rH7M2ZNQ1uZRm+8sY4WOFV71HI2rs/7y/lB7fdFYbawv8AfJ/GmNrA4Jf8zS9ox+zNlm1F8A6hKpA9VA/lSt9uTGNTmwT0EmKwW1hBklhUf9tQq33x+dT7SQezN11uJFAe/mz6mds/zqu2lxMxZ7kk9/3hz+dZDa/F0DD86RvEES9HFLnkV7M0m0Kz+8+x2I+8VzSrpVkvZeueExxisiTxJF/z0A5qJvEsfGDkUnKQezN06bbMu4HbjouKebK2+U4YnvkCucbxMvTk1G3ig9hS5pFezOk/s6LacdM1Iun2v8TMvpXJt4mfbkBqhk8RSsPumlzMfszr/sdn3LGm+TaL9wZ47nFccNelZRhDUba1cH+E0XkHszsvLtVyfLBGPXvUe2EqAI1x681yDatecYBxSf2lfFe+KV2x+zOvVolP+rQ49c083EK4P2e2/wCBbj/WuL+13rHoc+lKz37YJDAdOlLXuV7M69rxZDjbbRLn+CLn+dRtd/P/AKxML6IBXJrHqEnOGx2qVNI1idvkhlYf7Kk/you+41TOmbUtykeaB7KopjakRz9oP1wKx4fBXiS7/wBXp15If9mJjWhB8JfGN0Mpoeot/wBu7/4VHPFbspUm+hN/bhj5FztPqqiq82sed/rLyUjsAwA/lWlB8A/Hlx93w9qBGccxEVpW/wCzH8Q7gjHh+8BP95cUvaQ/mK9k+xybalB0Mjt9XNH9rWq/xc/7xP8AWu9t/wBkT4jXBydClQHuzgVr2v7FPxEucf8AEuhjH+3cIP60vbUv5h+xl2PKH1q0bOeR9c/zqFtYtOTtT8h/hXu9p+wd43uMGWSwtx33zg4/Kti1/wCCfPiJlHn65pkZ74Lt/wCy0nXpLqP2L7HzY3iCJPu7fpimr4nK8Bsj0r6qtv8AgnveBf3/AIntc99kDGtC3/4J9wK373xLkf7Nsf6tWTxFLuV7CR8mR61HdDZIm9H4INev/Bf4cal461q1sLeOQ2aPmS6ZSViTrgnpn0r3vwz+wt4d0e+jn1DVJ9QjXkxBAgP1PWvoXw34V03wpp0VjpdnFaWyDASNQK5a2JT0gdFOjbVlbwv4VtfDOlwWlugVIlC+5xW75WKmC+1DLXmnWRbaUrUm2m7TUAMxTcelSkU3bQAzmj9adtowaAGjvR0J606jBoAbtopwooAi20Faf+NLWohqqTTtlAB5FSLnvRYYwR880/y/SnCpMZqQK+3rVW8tfMjPrV9kLUjKdtAHAa1Z4VsDJry/xda7Y2bafWvddY00SxsyjnvXlnjDTdsMgxxS0GfMHxMj/wBOsn9Y2XP4iuCkTk16N8UYdv2Ug8q7qf0/wrz+RT3r6DDv92jx6699lRl6Vf8AD641SAkZXdzzVZkLAelb2iwwWdlHfTSAozkDaM8iu+Lujm21NuRCucDjtTF3HtxU/wDbGlSKCbnfx0RScVGdc0uNSFkZj0xsNLlNPaIjZj6UMHBA2mlXXtPC/ckI/wBzFObxJYlvkt5sAei/1NFifaIgKOxzg8UeTIzYCkmnt4kh6x2cjHH8RVR+maZ/wk0q426egHq1wT/7JTF7QX7LMp5Q/lT/ALLIyn5Tmmf8JFdMpxbQrz/FuP8AUUNrV0qqUigX+9wTn6c0WF7QPskzdFNL9hmYg7T+FN/tnUmXIEA9zED/ADpZNY1Uj/XIg4zthUfyFMn2jJRpcz5wMUq6TNkAj9Koi41CRgGv7gc54bAprSTbmDX0re++jQn2jNRdIdW+YNj6U7+w5JOB1rKPq95Lg/8ATUj+tQMlq2RJcytz/E5Ip2Qe0kbzeHJlPzER5/vECm/Y7W23LPf2sR7750GPzNc8trpfmAsisOpOOank/s6HhFjYdvLU0aC5pGyZNL4H9o2pI9JQ38qVW0pQN14uf91iP/Qaxf7SsATi3OR0wOv5ml/ta2TgWy/kKV0K8zXe60fvcNj/AGYm/qBUbaloyY2x3sv+7Bwf/HqzW8QRqq7bdOPUUxvEzKwxDGB7UcyF7zNX+1LFcGPSrx1bIDNhQf50q6uzHEOhzMcE5aRccfhWRJ4omZcbEHOaibxRdrnbIqg8fKKfOLlkdAt5esgcaGoGM5klxUkdxqjf6nRrQH/amPH6Vy0nia76eecenakbxFc+WV+0MAxyfej2jDkZ1puNfbG2z02PPdiSB+ooZ/EvllhdWFsuOTDFk/qTXGNrk/8Az9OP91qhOrPzmaTn0c0/ayD2J262+sNy+vQrnriGP/4movs96x+fxDMOwEexQf0rh31IMeXZvQljUL369TtP1pe0kxqkjvmhdGxJ4jvgo6hbk/yBqG4tdNfPm6zdXXTPmTMSeOc81wbamka/KVX6Ux9YTuw/OlzyH7M7WWz8NKrFgzOemGb/ABpsf/CNxqQbZX6ckEmuJbWol43801tciGfnpXkUqZ3C3fh+NsjTkkG7PzLnj6ULrGiqpC6cMLnHygVwTa9F/epja/GD1pNsfsz0CPxHYx8fYYx9FBNL/wAJdFDkR2keOvIFedHXlbPNMbXx0AJpq7H7M9GfxtKfuW0SfLtzwT/Liom8a3xYlQihh2b/AOtXnba869FOai/tyVjwP1osx+zPQf8AhLtQByHjUem3/wCvUM3iu9k+/MBzngYrgjrE3Xb+tR/2tM2fl5pah7M7tvE143/LwQOw9KRvEFw+C9y5IHrXDf2hcsOFNI11dNxz+FOxXsztW1qZutzIR6buKa2rHPMzE+5rila8Zv4qd5F43Teal+pXIdg2rKB9/P41C2rAkHf+tcuum30rYCSE+wNTx+G9WlPy2tw3PZDSuu4/Z+RuPqyr/FzUTavGP4wPxqlF4G1y4YBNPumP/XJv8Kvw/CXxVdf6vRL9x6rAx/pS5oLeQ/Z+RA2tRd3WmNr0A/iFbtv+z/44uceV4a1B8/8ATEgfrWnB+zD8QbjBHhu7UH++Av8AWsnUp/zFqiziT4ij/vfTNMbxGnqfyr021/ZF8f3GCdGC/wC/MgP862LX9i3xzccvFZwf78/+FS69JdSvYy7Hiz+IAzd6ibxAW/vZFfQkH7DfiqTHmX2nx+v7wn+ladv+wjqzAebrVoh77QTUvFUV1LVGXY+Y2159pwppn9vTHOFNfWVv+wfKP9b4gj+girRt/wBhTT1wZ9ekPrsiqHi6Pcf1eR8df21P0ApP7WuWxhea+3bf9hvw1HjzdZvJB/sxqP61rWn7FXgeH/WT6jL9HVf6VH1ymP6uz4K/tO6Pt6Un9oXfb+VfoVD+x78PIQN1ney4/vXGP6Vq2v7Kvw6hUf8AEjaUD/npOxqfr1PsV9XPze+0XkgPJ/Cjde991fppb/s3fD62YFPDVsT/ALRY5/Wta1+CHgm3H7vwvp47fNAD/OpePj0iP6v3Z+Wy2182DiTH0NSx6Pqc+NsMzfRTX6q2/wAK/DFrjyfDmnJjpttl/wAK04fBumQ48vSrSP6Qr/hUPH/3R+wR+UEXg/W7rhLG6b3ETf4Vaj+Gfie44j0m9b/ti3+FfrAnh+3jGFsrdP8AdjUf0qRdJEZ4iQfRRUfX5divYI/KyD4LeMLrATQ75/8Ati1aUH7OPjy5+54dvju/6ZGv1IXTnXpgZp66e55zUPHVOw/YxPzLtf2UPiHdMAvh64B6fNhf51q2/wCxj8Rbj7+lLEf+mk6DH61+kH9mvT1sWX72SPSp+u1B+wifnbb/ALDvj2ZgHjsYh3LXI/wrTg/YQ8YSAeZfabCO5EjN/Sv0EXTQ3fj0p39lY5GfpSeNq9w9jA+DLX9gfXWx5utWKf7qsa1bX9gKfjz/ABHED32wE/1r7e/s8KeVp/8AZqNzipeKqvqV7KJ8awfsC2AC+b4ic+uyHGf1rRt/2DfDyEeZrN049kAr66XTVHSnizC8YqPrFTuP2cex8q2/7DPg2P8A1l5eyHvyBWrbfsU+AY8b1vJT/wBdQP6V9L/Y1x93NL9jUYwP0pOvUfUORdj59tf2P/h1bEE6bcTdsPMf8K1rX9l/4cW448NRS4/vyOf617b9lA7Un2QZ9qz9pLuPlXY8ntv2fvAFp/q/CVh/wNS38zWnB8H/AAfbf6rwtpi/S2T/AAr0f7MvpSeSM1PNLqx8qOIt/h7oFv8A6nQrCP8A3bdR/StCLwrp0ONmn20f+7Ev+FdP5antzS+SF7VXM+47GAuh26/dt4l+iCpF0sLwEVR7CtzYPSjYO1TuMxf7Paj7CdvWtvyx6UnlD0o5QMb+z/zpRYFeRWt5INNaHmnZjRnrZ+tPFitXNlLsFPUTKX2FfSj7GPSrvTtml2g0wM9rMelM+y+grRaP2pPLpAUPs/qKY0Az0rQMdMaOiw9Sh5XtmmmOrzRimNHUjKLR0hjq20dNMeKkCp5eKNlWfLppTmgCvsFJt9qsGPHXpTdlAEOBRU3lmigCtSnFKqnrS4rURHnBqTIpu0/jUiIO/WgLirUgUZ5pFFSbaBCeXjvmjyxn1qRVp2MilYClNbhu1cP4y0EtC7gZH869F2hl5qpe6el1CyMMgjvSsM+FPi9YG2UgrgrJn+leSsGc7UG419P/ALQ3g2azYuExHJwHxxXzNq1uY99t5jRL0dkOGPtXsYWS5LHn4iL5rmJq0z3Ug06zfdO3E0y9EHp9a3bGyTS7eKzdmmWLoGORz1rPguLXS4PKgRUHc5yT9TUUmuIDnOTXoc3Y5XFtG/mNVG0YHsKadgzla5068Ox4qNvEHBFVzMz5DpvtEf8Acz9TQt0itxGCPQmuSbXjnADU1tclxwppXYezOyF8q9geMUx74buMAfTNcYNbnPQH8aZ/adzuzg4palezO0k1QYUK3QcggYzUf9rHbgvkfQVxzahdf3ODSC4vJFyAfwFPXuPkOsbUzjBkb8DTf7T7byc+9cn/AKYx6nNO+y30nZjT+YezOnbVBnlzn/eqM6pEAfmH51z0ei6lMflgmf8A3VJq7D4J1ub7mn3TfSJv8KWncfs32NFtXi6lhUT65EOARSw/C/xLcKCmk3jA/wDTJv8ACtK3+BfjK6H7vQb9/wDti1TzQ/mGqb7GO2uqf4wPxqNtfTGN/Suytf2Z/Hl3jy/D93z/AHkxWzZ/sh/EG4Uf8SlY895J0X/2ap9tSW8ivZS7HmJ8QL13E+9MbxIpyMEn6V7TafsV+OpP9bFYxf710v8ATNadv+w74pbBkvdNiPf52b+lP6zRXUr2Eux8/t4gyOlM/t5ugU19L2v7C+qt/rtZs0P+wjH+da9p+woBjz9fXHcRwf8A16zeMo9x/V5dj5P/ALak/uk0z+2Jm6LX2ba/sN6KmPO1u5Y/7Eaite1/Yn8Ixj95fX0p9torL67SvoP6vI+GTqdyx+7R9suPevvq3/Y48DxEFlvJf96TH9K1Lf8AZN8A2+M6dNJ/vSml9eh0RXsGfng012+CCRS7bxuBuNfpJb/s0+ALcZHh+Nz/ANNGY/1rVtfgZ4Js8eX4YsSR03pu/man6/H+UpUO5+YotL2TI+epF0fUJuQsh9gDX6m23wz8NWv+q8N6ao/69Ez/ACq/H4O0uHAi0iyjHbbboP6VDx3ZD9gflTF4V1eY/JZ3EnptiYmtG1+GPiS7YeXo9++f7sDH+lfqgnh+BB8lrEn+6gFWI9GA5ESr+FR9fl0RSoLufl3b/A3xldEeX4evzn1hYfzrUtf2afHt1jGgXQ/3gB/Wv03XSmPbFSDSW29Kl4+p2GqK6n5s2/7KPj2bGdHZP95xWnb/ALHfjmc/NZwxZ/vyiv0T/scjkUq6TurP67VKVGJ8A2v7Evi6bb5ktnF65kzWvbfsL663+t1awh/76P8ASvun+yMYINPXSw3Wo+u1u5XsonxVbfsIzbQbjxFbg9wkDH/CtS2/YU03/lr4imP+7bD/AOKr7DXSlx0zT/7KT0Gfes3iqr6jVOK6HybafsO+G48CbWr+X6RotbFv+xd4KhYF5r+X2Mij+lfTh0xOBtB96cunqv8ADUvEVX9ofs49j54tP2Rvh/bth7G4lP8AtTn+mK2bb9l/4f25G3QVf/ekY17mtirLyAaetmq9F4pe1m92PlXQ8dt/2f8AwNAMR+G7Uf7y5rTt/g54TtcbPDtkp94ga9T+xqeelL9jXjI/GpcpPqHKeeW/w30CDmPQrFT/ANcF/wAK0YfB+nQ/6vTLWP6QKP6V2S2qr2p3kA9qVx2OXj0VIRtSGNP91AKn/s2TbgcD610P2cDtThbjjjii4WOXbR369/WlXS23YNdSIB6VG1qDyB+FQUc5/ZJj5yW9akXS1+8M1u+Xjgil8gdhildjsYo0oUf2SFOcVtiEHrTvLBoCxirpkbDO2nHS0/u1reVtpfLFAzIj05c/dGKk+wJ/drSaDd0pNueDQBSFio7Ck+wp1ArQ20vl98UCKK2intTvsar/AA1b8vHNLt70DKn2dehFBtx24q3t79qT2oFoVhCCemCKd5A9BVjb7c0ipRYdyv8AZgvOKUQj05qwy+tIV54p2Fch8r1FJ5I9Kn69qAvqMilYLkAixUigdKk2+g4o21XKIaYwaY0W3GKk5HWn8Yx1p8oEG2hl9RU+0GmGMj3FHKBH5foaFB71JSdetKwDStJtp230o+oquUBuKTbuqTjHNH05p2AiVMU8KKX8MUhFPlAbsApGjp44petOwEOKGzUhFG2iwiI0nbFS7c0nlijlGRYpNoqQrtpMCiwERXH0pDUvbpUbLSATp9KOtHOaTjOM80gAr3prCnUN2oHuRlKbtqQimkVIyFk/CmMhqfbSFaQFbbQVqwVqMrntSAhK00r7VMVpNtICAr+FFS7R6UUAUxTdvNOwaXbitiRNvekOVqTbS7aQAnzDOOalVCB70xAc1Oq4NACIuOtSbc0gXmpFX2oAZtwfal29ak8vOMU5Y+vFAHOeLvCOmeLtLkstSg86JueDgg+oNfPPif8AYx0e/kebTtYvLd2JOyXDr/LNfU00OaptaktWsJyg/dIlFS3PhfXv2M/Etnk2M0F+uePmKt+tZtv+x34yuFy0FvCCON83I/DFffi6f61MuljNb/Wavcy9nE+ELX9iXxNJjzLuzh+r5rXtf2GtTkx52s2i+oVWNfby6WpFOGkp2FDxVRrcfs4nxfD+w2c4l16If7sJP8607f8AYf05SPN16Vv9y3H+NfYK6SmPugn1oGlp3WsXWq9ylCPY+VLP9ijwxH/r9TvpD/sKi/zBrZg/ZA8BWcebj7dKo6tJcKg/Ra+lP7LVuoxUv9npt2lFYehGaj2tRdSuSPY+f7P9lf4eRqGTSJJl7M1w7A/rWlH+zd4Dt+U8OQEjn5tx/ma9xXToxgKoC/3VGBS/Yk9Kl1JvW4+VHkNp8D/BsOCnhqyyPWEE1s23wu8OWgHk6BYxn2gX/CvQ1tVV8gVMtqPSo55dWVyo4aPwVp0X+r0u1jI9IVH9KuQ6DEnAtYgfaMD+ldd9mX0pWtQT0xSu+4rHMro237sar9BT10tx6V0ccQ3bSOad9nX0qRnNrpRVjxipP7IZuh5rohCAvAoWPtigDAXRy3fGKeNHzxW/5PfHFL5e7tTAwF0cKRxxUq6SvcZraEI9KPLA4xSAx10lBnC8VMNNQ4+UZrT8rinLF6inqIzRpqMOVpw09RjAFaXlj0pfLH407CM37Dt7Cl+xp0C1o7fWkMfcUDM/7GF/hWkW0A7Cr7IeMAY70vl/NilqMpC1FKLdegFXPLo8uiwFXyAO1P8AJ+XpxU+3NG2nYCDyR2FI0AC571Z25pGXPSiwiqIxStGGHIqw0YpPKwfUUcrGVfL/ACp6wj6mrDKPSk2ntRYBnlgj0pPJ7Gpl+nNP20+UCt5JHSjb6VM1N20+UBm2nqu6gUvl96dgG7cds0u3PGKevHWnbe4NOwEXl8e1O2U8LntzRtP0osAzYKTy/XmpB+tGM0rARtEG4qLyyh+7VrbtpPvClyoCq0eBkdetIlWWTHuKjZMjI4NKwDGzjpR5e36Uoz0NPFIBuM1FJGV+brUpXbyKOop2Agj+bnP4VLj0prKV5FLu3UrADJnFIVPpUmaQ9KYCDFIYx+NGBShsGnYBOfSl7elFJnHWnYBMetJjHWpBRTsAwruNAHrTsYpSMj0osA2j6Ufd60vWqsIbj25ppUjpUmDSc07DGe9G7NDLupPwqAFIzTelKuRS0wG5pPqKew9KbTAb06dKA2R6Uu38qbtzz0NACk5ooFLxVCEz60mcU7FNIIqgDdmjrR2pqg0rBYdRSc0e9AhTiomAp5amH5utIY2kzQw9OlJmpGNakz6ink0lSMbx9aRuKd0pDg0gQ3g02lIoqRhTWp1JmgBmKbT6jbOM1ICH3pjVHcXkVvGzyyLGijJZiAB+Nef+Ifj54K8Ps8c2rxzzL/yztvnJ/HpVKMpbILnoJb2or55v/wBsDS47lltNGuJYezyOqk/hRWvsKnYXMu572tP25zSx46GpQoOKBEW38aeI/apvLFO29KVgIlj+bNTKpNCrUqr3xRYBqx7akVefWpFSnrHRYBFjp/l96einHSpNvSgCFowVqIwruxjNXCue2Kb5dMRDHFUyoG7c0uOakX1oEIFHpUix0qqKeF6VVhCKvalCDPSn7T6Uu3ilYCPbR5fUnipNtG3FQ0UM27elJtyaf+lC8mosMjaOlTntg1I3vzUa/K2M5NKwEm0YoVaXndS0WAY0fzZxTtvfpTuaFG0nPSnYBNvrzSGM5xUuKOKfKBH/ADpdvcU7IzSj61XKK4m3NBj/AANKtOOaOULjdvY0Yx3pc9KT8aLCuLS+9IOlAquUQvHrRzSqu2l27qrlFcbt9OKTb1qQehpdtPkC5DtNJ61Nx0xTWUfWp5Rplf7vanK1K647VHk9uKmxRIBTqjWQ49qfuzQkK4mB3oxS0elVYLiFaXaRwRThS7crjNUogN2/hSU77vBowOvWr5SRCKbt9OKfnHWkIHanygNxx05ppyKfjNFTyjEGOtHSkxil4qQDk+1J97rS520dagA6deaWk/lRn0pDQ/tSfSm7jRkVJQEnpTWGOlLupjsR0G72oADhu3NJ04p3fgc0detIBjfdwBUe0hsk/lUjLg0vrQAlRlducGntkcilVgy8j86AGbs8Yp34UjLTR97rzSAdSFe9Gc0fjTAT+LBpeMUhw1AI70wAHbTsj1pKY1UgJPpRSKwowe1MQZB7Ucr24pc5HWjOO+KsVg+9TR15HFHuOtG7dwRikAm2mnFP579abtzUlDf5UdKXGBTffNADvxo4702lzQAYFGc9RilzR1oATFJt9qXHoaGJHbP6VQCY6GkJNKaQ57UyRCR3pcccU0+4o4p2GHSk5XFLtxSc0AJTWzT2pu4UgG4pu3BpxNNY9qljEYUlLmk4pANYUzO2m3F3Da482VYx/tHFZGpeMdE0s4u9St7f/rpIBUWZRs5A9qa1eQeK/wBpfwp4fuPIglfU5Rzi3GR+deV+Iv2w7/7VJHp+nR20C9Wnb5h+XFaxoTkrpEuSXU+sN3bIrnPEnxE8N+FFYarrVnZOozsklG78utfCfij9pDxFql1cznXZoV3Y8u2faFHoMf1ry/WfHN5qVyl4qSXUjHk3khJb3PNdEcI/tGbqpbH3Xr37W3hTT5DFp6XWpvg7Wjj2qfxOOK8k8RftkavcSXSwJYaWg4j+0Mxk/AZxn8K+Vb3Vb66yJJmhDHlICVX6dao3Ugm2Bo1Kx9Gxz+PrXZHCwRg6z6Hpniz40axrDySXuuXt60nKW6OQnPcjOMfhXEX3iq7ugiRiOGHqzKMufxNZRcAj5Mk+nFVnb+6M89K6VTijLmZNc3Ek0xZZ5kB/hDUVQaRtxoq+UXMfsXt6ZqdU3LwcU3+GpY+9fOnoiquBTttH8NPX7tACIKlVaYv3qnXpQA5VFSUxPvVKvWgBcfnT/pSUq9aBAaQ0tNX7tAxKcrbaRutOpolk6injjimx9qee1aIQv8NOAFNHenL92qsAFc03pxUv8NRSfeFQ0Aho/nR6UlZMsKjkyrbqk/iqOb7tSwJV+YA06mx/dFO/iFACr60p5HpSHqaeatIm4itSgelN/i/Cl71dhBQOaX1pP4hTsA6jmil7U7E3E2mgr6daWl70+ULiCl5pP46UU7CFBOKUA+vFN9aenerAMfSlzQlIPvUxCZo4pOwpG+7+FQMYWG7b/FjJFMde9St2Pemt978KzZZEPmp23Bpq/ep60gDnv0pQwWkao+4qgJ+lLTI+gpwrREsOo5o6Gl9KVvumqENYZptFKtOwBncM9KM5FC/eNOpWGR/jSd6Vu9IO1ZMoQtSfShv6n+dOHesmMbuPfijd6Gg/dNN/hqGUOVuxpSTTTSfxCkA/jvSfdzSdhRSYC8daTrRSmkAhx3pjcU7/AApOzUwEDcdaKRelLRYBN3rSNg9qSlH3RRYBufXrS5wOmaG+6KB92iwBkUh+al70nrTAM9qWg0i1SATbtHHNOoPSk/ipiYtFNWndxViQtJSr92mjqaBBntSN7Ujd6KTKQfWk+tFJUjENFLTaAFLdqXNN/hpR0qgHcdjRn15pvrS1QmLwwIpu0dOlOHWmUxXCkIDU7+GmnpQITBXoaN5FH8NMagBxcGkIpnrTCSGODikWPZePSq80yw53SKoxn5jisLxlczW+m5ileI8coxHf2r5W+L2uak0ksbahdGPfjYZmx09M1rGlzdRXPp7xN8TPD3hGzludQ1S3RYhlkRwz/kK8j179r7SlhlGjaXcXciDd5k3ypt6Z9TzXyhfTSfapP3jcqc8nmsHXLiWGxuPLkdMuoO1iOPSumOHh1MnUZ7X8QP2lfEniCzaErHpsEgyWhGGYemTXi2oeNF1CRnkvLi9uGIOJWJx7ZNcZcXEst0A8jOFXjcxOOBSQgLnAxzXZTpxjsYSkzcuvGF5NG0cMENuQNpYDLE9M5rHaS6vIZvNlkkZuXO/t2ph6ml+6rkcHDfyNbqKRk5NkcJjjjaIDjsAOp96GmC8Y6VE3Ei44qtH/AK4/WnYgnklZmJ6r2pqyrjBUnjk5H8qT+LHbNNUnB+uKLBcGuAvJUsh4HWoJPnVmQ4YdqW44mZR0z0ph/pTsFyMyEnPUmioZf9YaKLBc/9k=)

Slika 1: Digitalni sat sa mrežnom povezanošću

# Repozitorij s kodom i datotekama

Tijekom pisanja rada sam se služio Github-om i tamo sam spremao svoj rad. Sada je on javno dostupan za pristupanje ovom radu. U repozitoriju se može naći kod, datoteke za izradu tiskanih pločica, datoteke za 3d ispis, program za prikaz slike pomoću brojevnog niza, te ovaj rad.



Slika 2: QR kod na Github repozitorij

# Tehnologije i dijelovi sata



# CAD dizajn

Sat je sastavljen od 4 stranice kojima je jezgra led matrica u koju se šarafe M3 šarafi sa pločicama. USB-c konektor na dnu se pričvršćuje sa 3D ispisanim držačem. Dizajn je rađen Fusion360 programom.

Slika na kojoj se prikazuje tekst, zeleno, zid, znak

Opis je automatski generiran

Slika 3: izgled kućišta straga

Slika na kojoj se prikazuje tekst

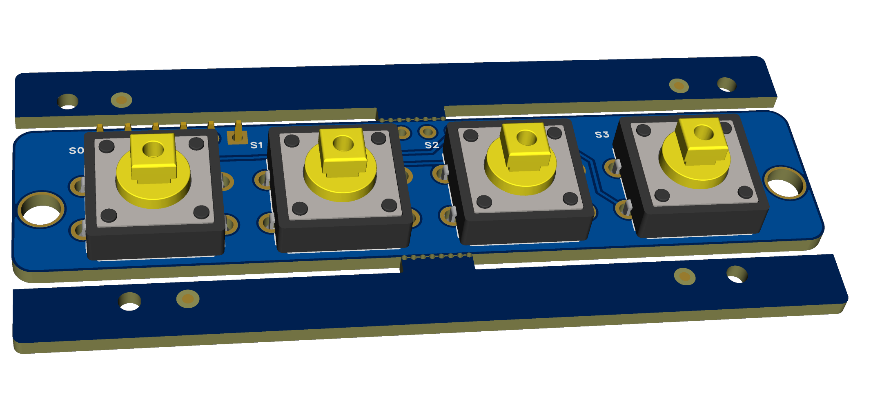
Opis je automatski generiran

Slika 4: izgled kućišta sprijeda



# Tiskane pločice

Tiskane pločice koje sam izradio su mala pločica za gumbe te velika pločica za ESP32 i proširenja. Pločice sam dao izraditi poznatom kineskom proizvođaču JLCPCB koji su mi ih povoljno izradili iz mojih dizajna koje sam napravio u EasyEDA Pro programu za projektiranje sklopova.

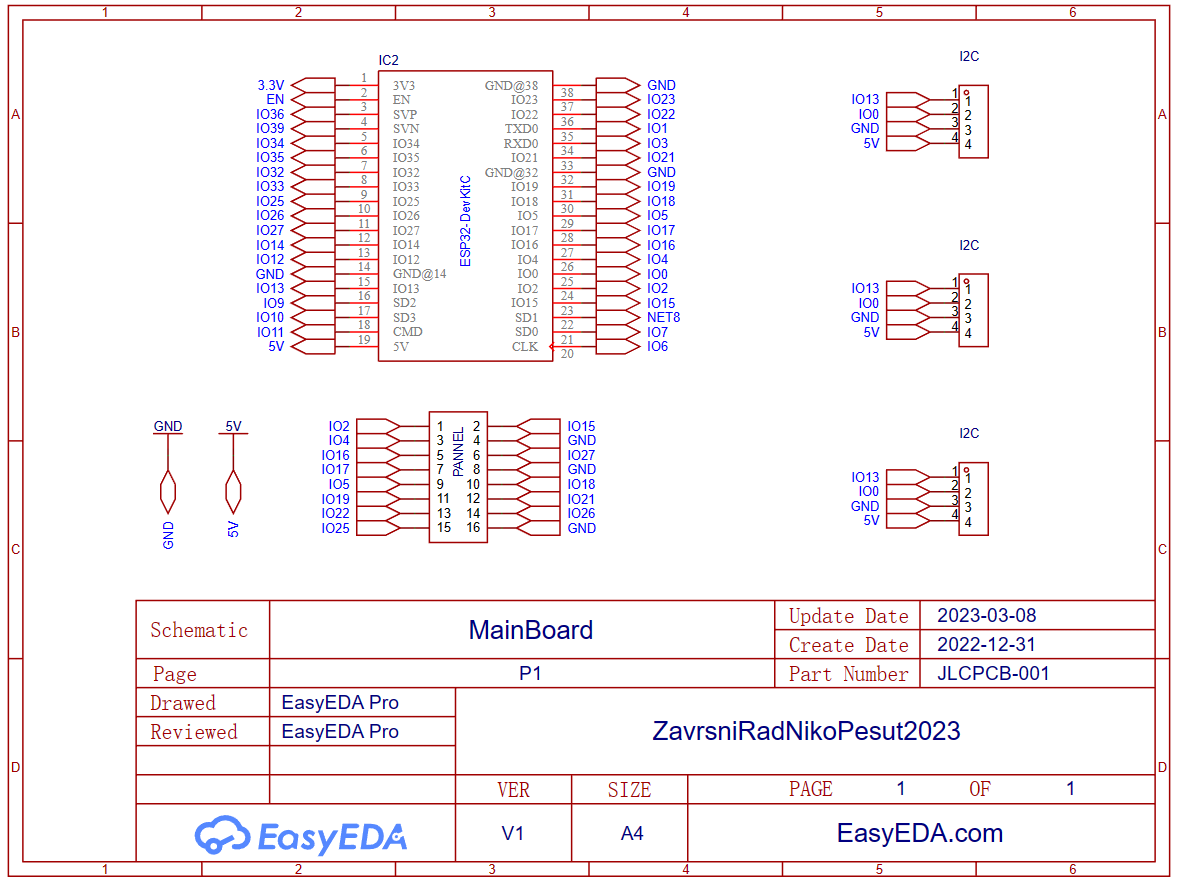


Slika 5: Pločica za gumbe

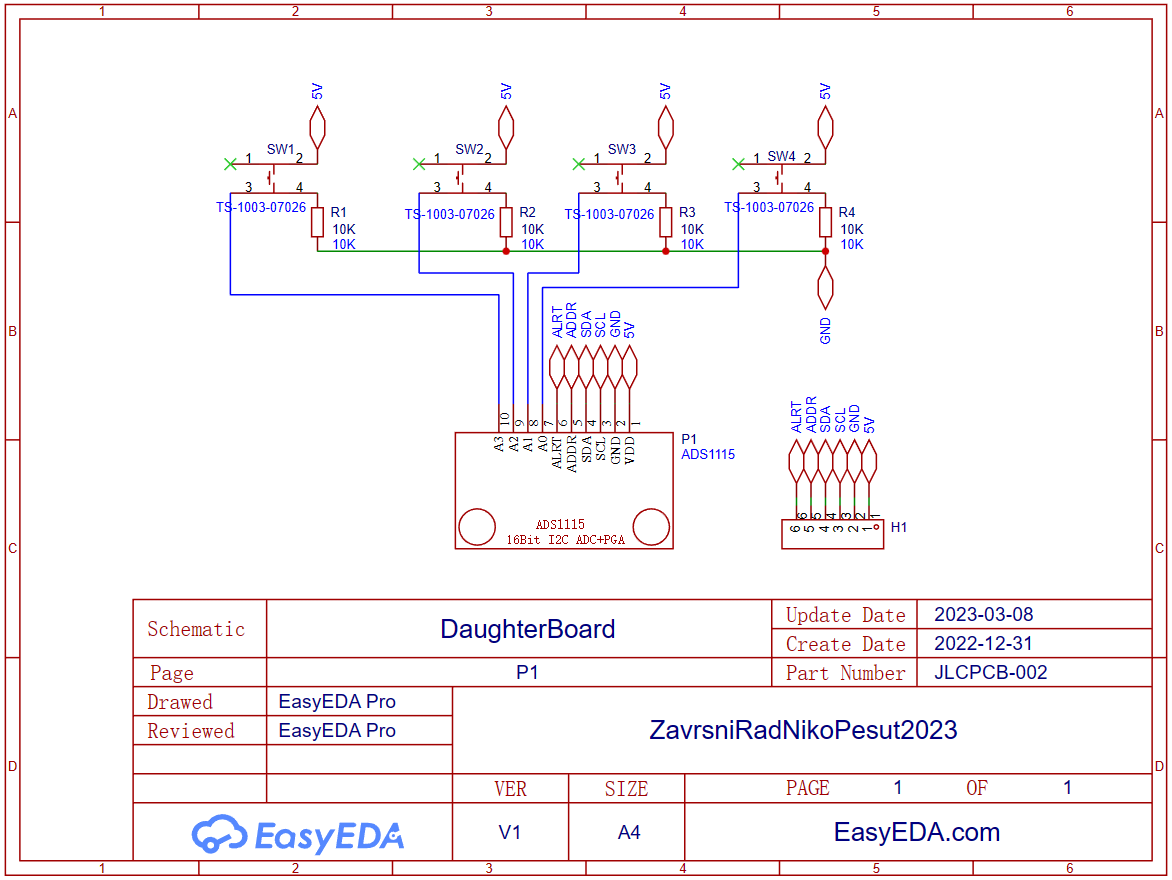
Slika na kojoj se prikazuje tekst, elektronički

Opis je automatski generiran

Slika 6: Pločica za ESP32 i proširenja

****

Slika 7: Shema pločice za ESP32 i proširenja

****

Slika 8: Shema pločice za gumbe

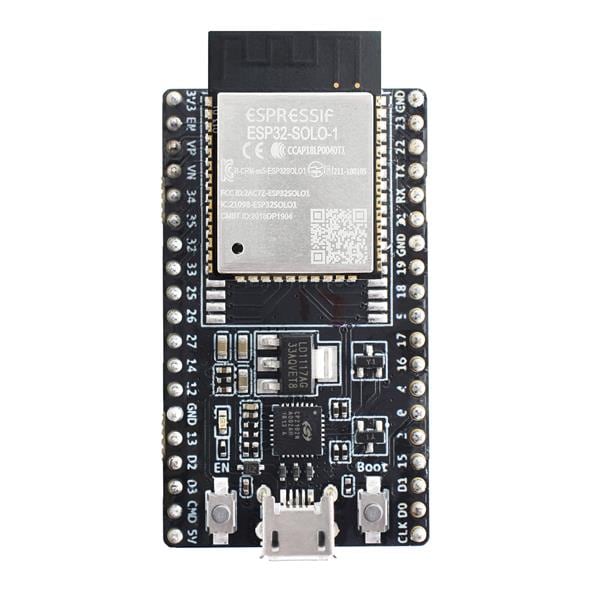
# ESP 32 mikrokontroler

ESP32 mikrokontroler je korišten u ovom projektu za upravljanje digitalnim satom. ESP32 je razvila tvrtka Espressif Systems, kineski proizvođač čipova za bežičnu komunikaciju, koji je poznat po svojim inovativnim rješenjima za IoT uređaje. ESP32 se sve više koristi u širokom rasponu aplikacija, uključujući pametne kuće, bežične senzore, automobilsku industriju, te mnoge druge.

ESP32 devkitc v4 verzija mikrokontrolera korištena u ovom projektu je omogućila višestruke funkcionalnosti digitalnog sata. Jedna od najznačajnijih karakteristika ESP32-a je njegova sposobnost povezivanja s WiFi mrežom, što je omogućilo povezivanje digitalnog sata s internetom i dohvaćanje podataka s API-ja.

Osim toga, ESP32 je korišten za sinkronizaciju sata putem NTP protokola, što osigurava točno prikazivanje vremena. ESP32 je također omogućio povezivanje s i2c proširenjima, uključujući DS3231 RTC i ADS1115 ADC, što je omogućilo precizno mjerenje vremena i druge podatke koji su prikazani na digitalnom satu.

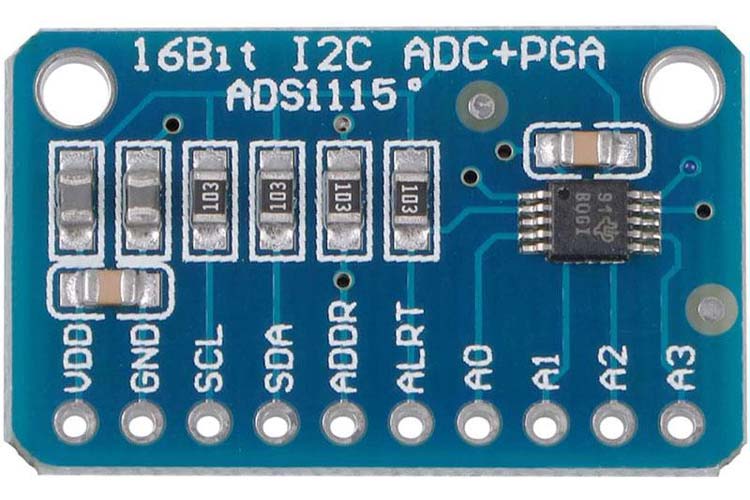
Naposljetku, ESP32 je korišten za upravljanje HUB75 64x32 matricom koja je prikazivala vremenske podatke i druge značajke sata. Sve ove funkcionalnosti ESP32 mikrokontrolera omogućile su stvaranje jedinstvenog i funkcionalnog digitalnog sata koji korisnicima pruža točno vrijeme i druge korisne informacije.



Slika 9: Prikaz sheme ESP32 devkitc v4 mikrokontrolera

# ADS1115 analogno digitalni pretvornik

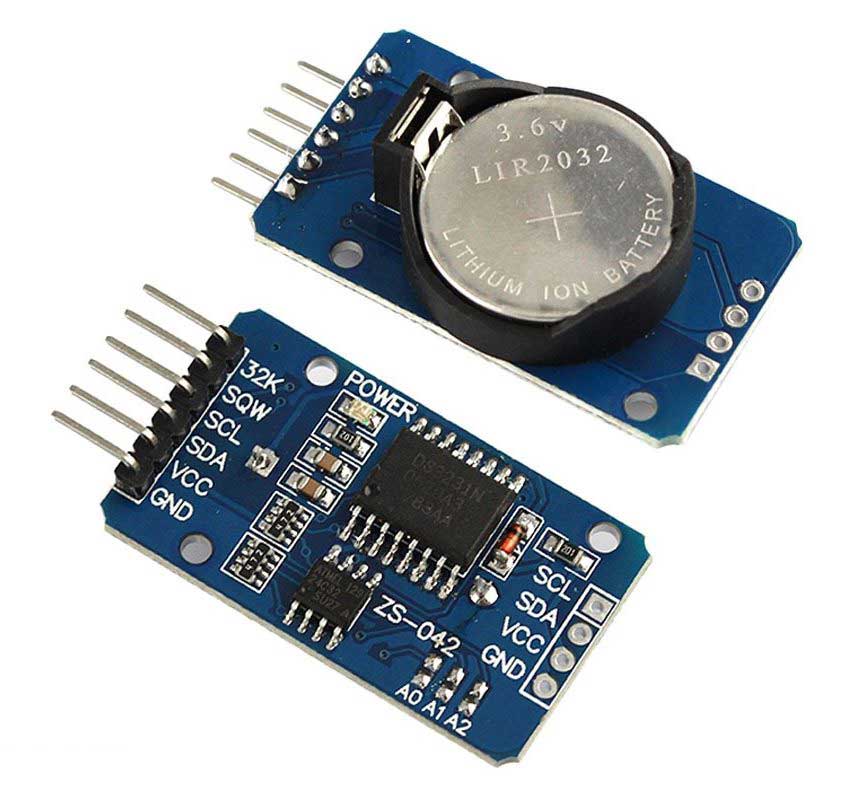
ADS1115 je I2C modul pretvarača analognog signala u digitalni (engl. ADC-Analog-Digital Converter). U ovom projektu, ADS1115 koristi se za detektiranje pritiska gumba koji se nalaze na pločici za upravljanje satom. ADS1115 ima četiri analogna ulaza i može pretvoriti ulazni signal u digitalni oblik na 16-bitnoj rezoluciji, što osigurava precizno mjerenje. Modul ADS1115 je vrlo koristan u situacijama kada je potrebno pretvoriti analogni signal u digitalni, kao što je u ovom slučaju kada se želi očitati pritisak gumba. Također, ADS1115 može biti koristan u drugim projektima kao što su mjerenje temperature i pritiska, detekcija razine tekućine i još mnogo toga. ADS1115 se lako integrira u projekt jer se komunicira putem I2C protokola, što omogućuje jednostavnu povezanost s mikrokontrolerom. Moglo se koristiti i U/I proširenje (engl. I/O expander), ali nisam našao niti jednu verziju koja podržava I2C komunikaciju.



Slika 10: Prikaz ADS1115 ADC-a

# DS3231 sat stvarnog vremena

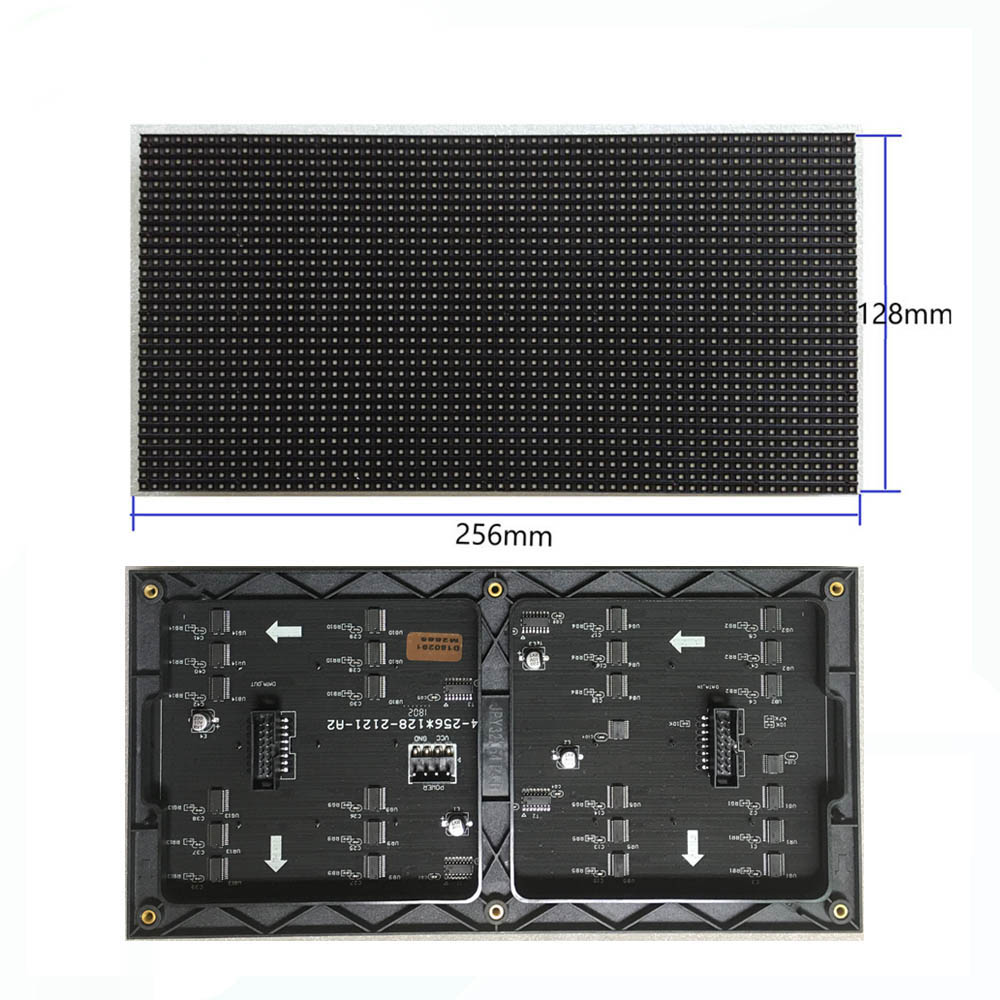
DS3231 RTC (engl. Real-Time Clock) je I2C modul koji se koristi za točno mjerenje vremena i datuma. Ovaj modul ima visoku preciznost i pouzdanost, što ga čini idealnim za korištenje u aplikacijama gdje je potrebna točnost vremena. U ovom projektu, DS3231 RTC modul koristi se za održavanje točnog vremena sata i kalendara. Mikrokontroler (ESP32) periodički čita vrijeme i datum iz DS3231 modula putem I2C komunikacije, te koristi te informacije za točno prikazivanje vremena na HUB75 matrici. DS3231 RTC modul također omogućuje automatsko održavanje ispravnog vremena u slučaju prekida napajanja, što je vrlo korisna funkcija u aplikacijama koje zahtijevaju točno mjerenje vremena.



Slika 11: Prikaz DS3231 RTC-a

# HUB75 64x32 matrica

HUB75 64x32 matrica je LED matrica visoke gustoće koja se sastoji od 2048 pojedinačnih LED dioda, raspoređenih u 64 stupca i 32 reda. Ova matrica može prikazati slike i tekst u visokoj razlučivosti i koristi se u različitim aplikacijama, uključujući LED reklamne ploče, stadionske ekrane i digitalne satove. Matrica se kontrolira pomoću upravljačkog IC-a koji se nalazi na matrici, a podaci se prenose putem paralelnog sučelja koje se zove HUB75. Za prikazivanje teksta ili slike na matrici, potrebno je slati odgovarajuće nizove podataka na HUB75 sučelje, koji će potom biti dekodirani i prikazani na odgovarajućem mjestu na matrici. Za upravljanje matricom u ovom projektu, korišten je ESP32 mikrokontroler, koji se povezuje na matricu putem HUB75 sučelja. Pomoću ESP32 mikrokontrolera, moguće je lako kontrolirati matricu i prikazati različite slike i tekstove na njoj, što čini ovaj projekt idealnim za prikazivanje vremena i drugih korisnih informacija.



Slika 12: Prikaz HUB75 64x32 matrice

# Funkcije sata



# Postavljanje konfiguracije

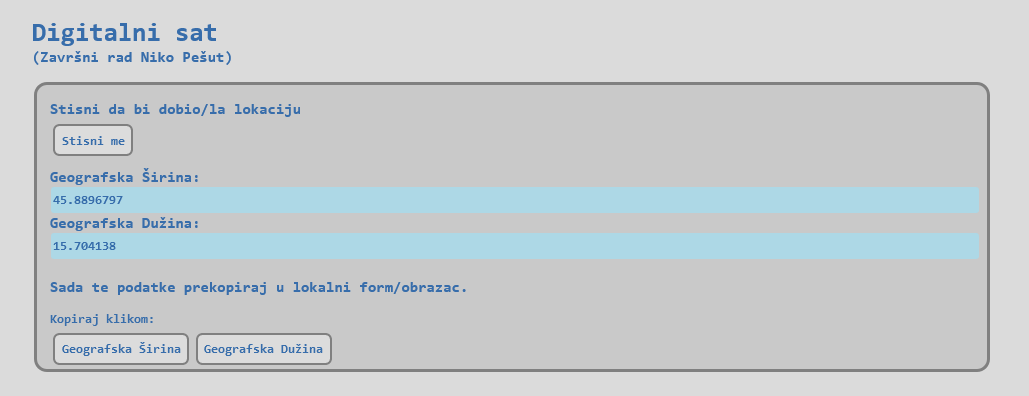
Konfiguracija se postavlja tako što se pristupi satu putem web preglednika na adresi 192.168.1.4, ako je upaljena pristupna točka sata, a ako je sat već spojen na mrežu onda konfiguriramo spajanjem na IP adresu koju možemo dobiti pritiskom na prvu tipku na vrhu sata.



Slika 13: Prikaz postavke konfiguracije u web pregledniku

# Dohvaćanje Geo lokacije

Geo lokacija se dohvaća tako što se prilikom konfiguracije pritisne gumb „dobij lokaciju“ i onda se otvara stranica „geoloc.pesut.win“ na kojoj se pritisne gumb i tako se dobije lokacija koja se može prekopirati na konfiguracijsku stranicu.



Slika 14: Prikaz dohvaćanja Geo lokacije u web pregledniku

# Dohvaćanje vremena

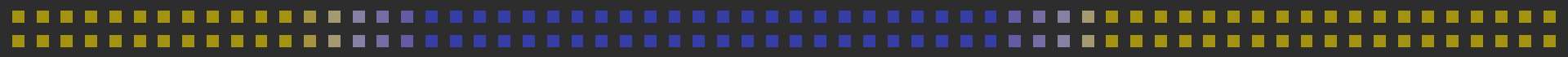
Dohvaćanje vremena se odvija tako što se zahtjev s zadanom geografskom širinom, dužinom i API ključem šalje besplatnom API-u (timezonedb.com) s bazom podataka vremena koji nam nazad šalje vrijeme (engl. UNIX timestamp) u obliku XML dokumenta (engl. eXtensible Markup Language) sa specificirane lokacije koje pohranjujemo u DS3231 RTC modul. Dohvaćanje se odvija svakih 5 minuta ako je sat povezan na mrežu.

# Dohvaćanje vremenskih stavki

Dohvaćanje vremena se odvija tako što se zahtjev s zadanom geografskom širinom, dužinom i API ključem šalje API-u (openweathermap.org) koji nam vraća JSON dokument koji sadrži trenutačnu temperaturu, vlagu zraka, vrijeme izlaska i zalaska sunca, trenutačnu brzinu vjetra, tlak zraka, vidljivost te pokrivenost neba oblacima. Dohvaćanje se odvija svakih 5 minuta ako je sat povezan na mrežu.

# Vremenska crta

Vremenska crta je prikaz izlaza i zalaska sunca, a dobiva se tako što putem podataka dobivenih od API-a preradimo i funkcijom uračunamo trenutačno vrijeme i nacrtamo početak crte kao trenutačno vrijeme. Ona se nalazi u donjem dijelu sata i služi da bismo vidjeli trenutačni položaj sunca na nebu.



Slika 15: Prikaz vremenske crte u 16:30 sati



# Pohrana podataka

Podatci se pohranjuju na internu memoriju ESP32 mikrokontrolera putem SPIFFS biblioteke, a pohranjuju se u obliku JSON (engl. JavaScript Object Notation) datoteke. Podaci koji se pohranjuju su: WiFi ime i šifra, ime i šifra pristupne točke, stanje matrice (prikaz), boje, intenzitet, geografska širina i visina, te API ključevi.

Slika na kojoj se prikazuje tekst

Opis je automatski generiran

Slika 16: Prikaz strukture JSON datoteke

# Tipke sata

Na pločici sa gumbima, postoje četiri tipkala.

Pritiskom na prvu tipku prikazuju se detalji o mreži, kao što su trenutna IP adresa i SSID mreže ili pristupne točke ako sat nije spojen na mrežu.

Pritisak na drugu tipku omogućuje prolazak kroz tri zadane boje za matricu. Korisnik tako može prilagoditi izgled sata prema svojim željama i potrebama.

Pritiskom na treću tipku mijenja se intenzitet svjetla, s pet različitih svjetlina koje su dostupne. Ovo je korisna funkcija, posebno u uvjetima smanjene vidljivosti.

Pritisak na četvrtu tipku omogućuje korisniku da promijeni način prikaza na satu.







# Prikazi

U nastavku su navedeni prikazi sata u nakon pritiskanja četvrte tipke, te prikazi pritiskom prve tipke ili tijekom pokretanja sata za prikaz detalja veze na WiFi.

# .

Prikaz 1 sadrži: sat, vremensku crtu, trenutačno vrijeme i WiFi status.

Slika na kojoj se prikazuje tekst, semafor, na otvorenom

Opis je automatski generiran

Slika 17: Matrica u 1. prikazu



# .

Prikaz 2 sadrži: sat, vremensku crtu i trenutačno vrijeme.

Slika na kojoj se prikazuje tekst, semafor, na otvorenom

Opis je automatski generiran

Slika 18: Matrica u 2. prikazu

# .

Prikaz 3 sadrži: sat, vremensku crtu i datum.

Slika na kojoj se prikazuje tekst, semafor, na otvorenom

Opis je automatski generiran

Slika 19: Matrica u 3. prikazu

# .

Prikaz 4 sadrži: sat i datum.

Slika na kojoj se prikazuje tekst, semafor, na otvorenom, biserka

Opis je automatski generiran

Slika 20: Matrica u 4. prikazu

# .

Prikaz veze pritiskom na prvi gumb, kada je WiFi spojen.

Slika na kojoj se prikazuje tekst, elektronika, ploča za bilježenje rezultata

Opis je automatski generiran

Slika 21: Matrica u prikazu kada je WiFi spojen.

# .

Prikaz veze pritiskom na prvi gumb, kada je WiFi od spojen i uključena je pristupna točka (engl. Hotspot).



Slika 22: Matrica u prikazu kada je pristupna točka upaljena.



# Kôd

Kôd je pisan u c++ programskom jeziku, te je kompiliran putem PlatformIO proširenja za Visual Studio Code.

Kôd se sastoji od 4 cjeline, a to su: glavni dio programa, konfiguracija i pohrana podataka, HTTP zahtjevi i crtanje po matrici. Još od kôda imamo HTML datoteku za konfiguriranje, te HTML datoteku za dohvaćanje geolokacije koja se nalazi na mojem Raspberry Pi web poslužitelju.



Slika 23: PlatformIO logotip



# Glavni dio programa

Glavni dio programa se nalazi u „main.cpp“ datoteci. Ona sadrži glavno tijelo programa i inicijalizira I2C protokol, matricu, DS3231 RTC modul, ADS1115 ADC modul, WiFi konekciju ili uključuje pristupnu točku ako se ne može spojiti na WiFi i pokreće task manager koji u određenim intervalima poziva funkcije.

# Konfiguracija i pohrana podataka

Konfiguracija i pohrana podataka se odvijaju u „config.h“ datoteci. U njoj se nalaze sve globalne varijable i sve postavke za matricu. Također se u njoj nalaze funkcije za čitanje i pisanje u JSON datoteku koja se nalazi na SPIFFS memoriji ESP32 mikrokontrolera.

# HTTP zahtjevi

HTTP zahtjevi se odvijaju u „HTTP.h“ datoteci. Zahtjevi se obrađuju putem ESPAsyncWebServer biblioteke koja asinkrono sluša zahtjeve od ostalog koda te pri pozivu se aktivira i obrađuje zahtjeve. U ovoj datoteci se nalaze i funkcije za osvježavanje vremena i vremenskih stavki putem API-a, te povezivanje na WiFi mrežu ukoliko se sat od spoji.

# Crtanje po matrici

Crtanje po matrici se odvija u „display.h“ datoteci. Ona sadrži sve funkcije za crtanje po matrici kao što su crtanje sata, broja, slova, teksta, vremenske crte i drugih. Također se može naći funkcija koja osvježava vrijeme sa DS3231 RTC modula i funkcija koja očitava dali je koji od četiri gumba pritisnut.

# Prilog Kôd-a

|  |
| --- |
| main.cpp |
| #include <Arduino.h>  #include "config.h"  #include "display.h"  #include "HTTP.h"  TaskHandle\_t Task1;  void setup() {    Wire.begin(I2C\_SDA, I2C\_SCL);    adc0.setRate(7);    if(!SPIFFS.begin(true)) { // Initialize SPIFFS          Serial.println("An Error has occurred while mounting SPIFFS");          return;    }    Serial.begin(115200);    Serial.println("JSON LOAD!");    loadConfig();    //for(int i=0; i<3; i++) boja1[i]=255;    //for(int i=0; i<3; i++) boja2[i]=(i!=2?255:0);    //for(int i=0; i<3; i++) boja3[i]=(i!=1?255:0);    //saveConfig();    matrix.addLayer(&backgroundLayer);    matrix.addLayer(&indexedLayerZ1);    matrix.addLayer(&indexedLayerZ2);    matrix.begin();    matrix.setBrightness(svjetlina);      indexedLayerZ1.drawString(1,1,1,"Wi-Fi spajanje");    indexedLayerZ1.drawString(1,26,1,(nName+"  "+nVersion).c\_str());    indexedLayerZ1.swapBuffers();      rtc.begin();    refreshTime();    myWIFI.begin(SSID.c\_str(), PASS.c\_str(), APssid.c\_str(), APssid.c\_str(), APpass.c\_str(), "192.168.4.1");    if (WiFi.status() == WL\_CONNECTED)  {      //mySSDP.begin(SSDP\_Name.c\_str(), "000000001", nName.c\_str(), nVersion.c\_str(), "NikoPesut", "https://kikirikiserver.hopto.org:60001"); Serial.println(F("Start init SSDP"));      Serial.println(F("Start SSDP"));   //Run SSDP      //myESPTime.begin(timezone, isDayLightSaving, sNtpServerName, "pool.ntp.org", "time.nist.gov", true, true);      clearAllLayers();      indexedLayerZ1.drawString(0,0,1,"Wi-Fi spojen");      typeText(0,7,(String)("SSID:\n"+(String)myWIFI.getNameSSID()+"\nIP-WiFi:\n"+myWIFI.getDevStatusIP().substring(9)),&boja2[0]);      refreshAPIs();    }    else    {      clearAllLayers();      indexedLayerZ1.drawString(0,0,1,"Wi-Fi neuspjeh!");      typeText(0,7,(String)("SSID:\n"+APssid+"\nIP-HOTSPOT:\n192.168.4.1"),&boja2[0]);    }    taskManager.scheduleFixedRate(300,refreshAPIs,TIME\_SECONDS);//svakih 5 minuta    taskManager.scheduleFixedRate(15,recconectWifi,TIME\_SECONDS);//svakih 15 sekundi    taskManager.scheduleFixedRate(1,refreshTime,TIME\_SECONDS);//svaku 1 sekundi    taskManager.scheduleFixedRate(100,displayDraw,TIME\_MILLIS);//svakih 0.1 sekundi    taskManager.scheduleFixedRate(99, gumbiSense, TIME\_MILLIS);//svakih 0.1 sekundi    swapAllLayerBuffers();    httpSetup();    delay(10000);    clearAllLayers();  }  void loop(){    taskManager.runLoop();  } |
| config.h |
| #ifndef GLOBAL\_H  #define GLOBAL\_H  #include <HTTPClient.h>  #include <MatrixHardware\_ESP32\_V0.h>  #include <SmartMatrix.h>  #include <WiFi.h>  #include <AsyncTCP.h>  #include <ESPAsyncWebServer.h>  #include "SPIFFS.h"  #include "NetCrtESP.h"  #include <RTClib.h>  #include <Wire.h>  #include <cstring>  #include <ArduinoJson.h>  #include <TaskManagerIO.h>  #include <ADS1115.h>  //#include <TimeLib.h>  //#include <Timezone.h>  //#include "ESPTimeFunc.h"  //#include "time.h"  //#include "ssdpAWS.h"  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*JSON\_Variables\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  String filePath = "/myConf.json"; //File for config  String jsonConfig = "{}";  String SSID="", PASS="";  uint8\_t stateID=4;  //String mqttPass = "";  //String mqttUsername = "";  //String mqttIP = "";  String APssid = "SatZavrsniNiko";  String APpass = "";  uint8\_t boja1[3]={255,255,255},boja2[3]={128,64,128},boja3[3]={64,64,128}, svjetlina=128;  float lat=0, lng=0;  String timeApiKey="x"; //MCD644HWLRP5  String weatherApiKey="x"; //2f940c078e7a96ecf846d11bba2cbdc7  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Gumbi\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  ADS1115 adc0(ADS1115\_DEFAULT\_ADDRESS);  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Serijska Komunikacija\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  const int I2C\_SDA=0,I2C\_SCL=13;  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*RTC & sat\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  RTC\_DS3231 rtc;  String daysOfTheWeek[7] = {"NED","PON","UTO","SRI","CET","PET","SUB"};  uint32\_t epochNow=0;  bool tocke=true;  //ESPTimeFunc myESPTime(false);  float vrijemeIzlaska=1671261493%86400/60.0/60, vrijemeZalaska=1671301493%86400/60.0/60;  int minuta, sekunda, sat;  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WIFI\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  NetCrtESP myWIFI;  AsyncWebServer HTTP(80);  //ssdpAWS mySSDP(&HTTP);  String SSDP\_Name = "sussyESP";  const String nName = "NikoSat";  const String nVersion = "v0.6.1";    /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*DISPLAY setup\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  #define COLOR\_DEPTH 24  const uint16\_t kMatrixWidth = 64;  const uint16\_t kMatrixHeight = 32;  const uint8\_t kRefreshDepth = 36;  const uint8\_t kDmaBufferRows = 4;  const uint8\_t kPanelType = SM\_PANELTYPE\_HUB75\_32ROW\_MOD16SCAN;  const uint32\_t kMatrixOptions = (SM\_HUB75\_OPTIONS\_NONE);  const uint8\_t kBackgroundLayerOptions = (SM\_BACKGROUND\_OPTIONS\_NONE);  const uint8\_t kScrollingLayerOptions = (SM\_SCROLLING\_OPTIONS\_NONE);  const uint8\_t kIndexedLayerOptions = (SM\_INDEXED\_OPTIONS\_NONE);  SMARTMATRIX\_ALLOCATE\_BUFFERS(matrix, kMatrixWidth, kMatrixHeight, kRefreshDepth, kDmaBufferRows, kPanelType, kMatrixOptions);  SMARTMATRIX\_ALLOCATE\_BACKGROUND\_LAYER(backgroundLayer, kMatrixWidth, kMatrixHeight, COLOR\_DEPTH, kBackgroundLayerOptions);  SMARTMATRIX\_ALLOCATE\_INDEXED\_LAYER(indexedLayerZ1, kMatrixWidth, kMatrixHeight, COLOR\_DEPTH, kIndexedLayerOptions);  SMARTMATRIX\_ALLOCATE\_INDEXED\_LAYER(indexedLayerZ2, kMatrixWidth, kMatrixHeight, COLOR\_DEPTH, kIndexedLayerOptions);  //const int defaultBrightness = (100\*255)/100;        // full (100%) brightness  //const int defaultBrightness = (15\*255)/100;       // dim: 15% brightness  const int defaultBrightness = (20\*255)/100;       // dim: 15% brightness  const int defaultScrollOffset = 0;  const rgb24 defaultBackgroundColor = {0x40, 0, 0};  #define BLACK   0x0000  #define BLUE    0x001F  #define RED     0xF800  #define GREEN   0x07E0  #define CYAN    0x07FF  #define MAGENTA 0xF81F  #define YELLOW  0xFFE0  #define WHITE   0xFFFF  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*DISPLAY.h varijable\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  int dan=0, noc=0;  bool gumbStanje=0, saveStanje=0;  uint8\_t zadnje\_stisnuto=0;  String statusTrenutno;  int statusTrenutniPomak=kMatrixWidth;  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*HTTP.h varijable\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  float tempCurr=0;  float humCurr=0;  float windCurr=0;  int pressCurr=0;  int visibilityCurr=0;  int skyCurr=0;  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* JSON \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  bool loadConfig();  bool saveConfig();  bool loadConfig() {    Serial.println("-----");    Serial.println("Load!");    Serial.println("-----");      if(!SPIFFS.exists(filePath)) {          Serial.println(F("Failed to open config file"));          saveConfig();          return false;      }      File configFile = SPIFFS.open(filePath, "r");      size\_t size = configFile.size();      if (size > 2048) {          Serial.print(F("Config file size is too large: ")); Serial.println(size);          configFile.close();          return false;      }      jsonConfig = configFile.readString();      configFile.close();      DynamicJsonDocument jsonDoc(5096); //4096      DeserializationError error = deserializeJson(jsonDoc, jsonConfig);      if (error) {        Serial.print(F("loadConfig() deserializeJson() failed with code "));        Serial.println(error.c\_str());        return false;      }      JsonObject root = jsonDoc.as<JsonObject>();      //network      SSID = root["SSID"].as<String>();      PASS = root["PASS"].as<String>();        APssid = root["APssid"].as<String>();      APpass = root["APpass"].as<String>();      //mqttPass = root["mqttPass"].as<String>();      //mqttUsername = root["mqttUsername"].as<String>();      //mqttIP = root["mqttIP"].as<String>();      //display      stateID = root["stateID"];      boja1[0] = root["color1Red"] ; boja1[1] = root["color1Green"]; boja1[2] = root["color1Blue"];      boja2[0] = root["color2Red"] ; boja2[1] = root["color2Green"]; boja2[2] = root["color2Blue"];      boja3[0] = root["color3Red"] ; boja3[1] = root["color3Green"]; boja3[2] = root["color3Blue"];      svjetlina = root["lightIntensity"].as<uint8\_t>();      //lokacija      lat=root["latitude"].as<float>();      lng=root["longitude"].as<float>();      //api-s      weatherApiKey = root["weatherApiKey"].as<String>();      timeApiKey = root["timeApiKey"].as<String>();      return true;  }  bool saveConfig() {    Serial.println("-----");    Serial.println("Save!");    Serial.println("-----");      DynamicJsonDocument jsonDoc(5096); //4096      DeserializationError error = deserializeJson(jsonDoc, jsonConfig);      if (error) {          Serial.print(F("saveConfig() deserializeJson() failed with code "));          Serial.println(error.c\_str());          return false;      }      JsonObject json = jsonDoc.as<JsonObject>();          //network mqttUsername mqttIP mqttPass      json["SSID"] = SSID;      json["PASS"] = PASS;      json["APssid"] = APssid;      json["APpass"] = APpass;      //json["mqttPass"] = mqttPass;      //json["mqttUsername"] = mqttUsername;      //json["mqttIP"] = mqttIP;        //display      json["stateID"] = stateID;      json["color1Red"] = boja1[0]; json["color1Green"] = boja1[1]; json["color1Blue"] = boja1[2];      json["color2Red"] = boja2[0]; json["color2Green"] = boja2[1]; json["color2Blue"] = boja2[2];      json["color3Red"] = boja3[0]; json["color3Green"] = boja3[1]; json["color3Blue"] = boja3[2];      json["lightIntensity"] = svjetlina;      //lokacija      json["latitude"]=lat;      json["longitude"]=lng;      //api-s      json["weatherApiKey"] = weatherApiKey;      json["timeApiKey"] = timeApiKey;      serializeJson(json, jsonConfig);      File configFile = SPIFFS.open(filePath, "w");      if (!configFile) {          Serial.println(F("Failed to open config file for writing"));          configFile.close();          return false;      }      serializeJson(json, configFile);      serializeJson(json, Serial);      configFile.close();      return true;  }    /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Podatci\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  bool brojke[60][13]={{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,0,0,0,0,0,0,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{0,0,0,0,0,0,0,0,0,0},{0,0,0,0,0,0,0,0,0,0},{1,1,0,0,0,0,0,0,0,0},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,0,0,1,1,1,1,1,1},{1,1,0,0,1,1,1,1,1,1},{1,1,0,0,1,1,0,0,1,1},{1,1,1,1,1,1,0,0,1,1},{1,1,1,1,1,1,0,0,1,1},{1,1,0,0,1,1,0,0,1,1},{1,1,0,0,1,1,0,0,1,1},{1,1,0,0,1,1,0,0,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,0,0,0,0},{1,1,1,1,1,1,0,0,0,0},{0,0,0,0,1,1,0,0,0,0},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,0,0,1,1},{1,1,1,1,1,1,0,0,1,1},{1,1,0,0,1,1,0,0,1,1},{1,1,0,0,1,1,1,1,1,1},{1,1,0,0,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,0,0,1,1,0,0,1,1},{1,1,0,0,1,1,1,1,1,1},{1,1,0,0,1,1,1,1,1,1},{1,1,1,1,0,0,0,0,0,0},{1,1,1,1,0,0,0,0,0,0},{1,1,0,0,0,0,0,0,0,0},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,0,0,1,1,0,0,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,0,0,1,1},{1,1,1,1,1,1,0,0,1,1},{1,1,0,0,1,1,0,0,1,1},{1,1,1,1,1,1,1,1,1,1},{1,1,1,1,1,1,1,1,1,1},{0,0,0,0,0,0,0,0,0,0},{0,0,1,1,0,0,1,1,0,0},{0,0,1,1,0,0,1,1,0,0},{0,0,0,0,0,0,0,0,0,0},{0,0,0,0,0,0,0,0,0,0},{0,0,0,0,0,0,0,0,0,0},{0,0,1,1,0,0,1,1,0,0},{0,0,1,1,0,0,1,1,0,0},{0,0,0,0,0,0,0,0,0,0},{0,0,0,0,0,0,0,0,0,0}};  bool miniLetters[285][5]={{0,0,0,0,0},{0,0,0,0,0},{0,0,0,0,0},{0,0,0,0,0},{1,1,1,0,1},{0,0,0,0,0},{1,1,0,0,0},{0,0,0,0,0},{1,1,0,0,0},{1,1,1,1,1},{0,1,0,1,0},{1,1,1,1,1},{1,1,1,0,1},{1,1,1,1,1},{1,0,1,1,1},{1,0,0,1,1},{0,0,1,0,0},{1,1,0,0,1},{0,1,0,1,0},{1,0,1,0,1},{0,1,0,1,1},{0,0,0,0,0},{1,1,0,0,0},{0,0,0,0,0},{0,0,0,0,0},{0,1,1,1,0},{1,0,0,0,1},{1,0,0,0,1},{0,1,1,1,0},{0,0,0,0,0},{1,1,1,0,0},{1,0,1,0,0},{1,1,1,0,0},{0,0,1,0,0},{0,1,1,1,0},{0,0,1,0,0},{0,0,0,0,0},{0,1,0,0,0},{1,0,0,0,0},{0,0,1,0,0},{0,0,1,0,0},{0,0,1,0,0},{0,0,0,0,0},{0,0,0,0,1},{0,0,0,0,0},{0,0,0,1,1},{0,1,1,1,0},{1,1,0,0,0},{1,1,1,1,1},{1,0,0,0,1},{1,1,1,1,1},{0,0,0,0,0},{0,1,0,0,0},{1,1,1,1,1},{1,0,1,1,1},{1,0,1,0,1},{1,1,1,0,1},{1,0,1,0,1},{1,0,1,0,1},{1,1,1,1,1},{1,1,1,0,0},{0,0,1,0,0},{1,1,1,1,1},{1,1,1,0,1},{1,0,1,0,1},{1,0,1,1,1},{1,1,1,1,1},{1,0,1,0,1},{1,0,1,1,1},{1,0,0,0,0},{1,0,0,0,0},{1,1,1,1,1},{1,1,1,1,1},{1,0,1,0,1},{1,1,1,1,1},{1,1,1,0,1},{1,0,1,0,1},{1,1,1,1,1},{0,0,0,0,0},{0,1,0,1,0},{0,0,0,0,0},{0,0,0,0,1},{0,1,0,1,0},{0,0,0,0,0},{0,0,1,0,0},{0,1,0,1,0},{1,0,0,0,1},{0,1,0,1,0},{0,1,0,1,0},{0,1,0,1,0},{1,0,0,0,1},{0,1,0,1,0},{0,0,1,0,0},{1,0,0,0,0},{1,0,1,0,1},{1,1,1,0,0},{0,1,0,1,1},{1,0,1,0,1},{1,1,1,0,1},{1,1,1,1,1},{1,0,1,0,0},{1,1,1,1,1},{1,1,1,1,1},{1,0,1,0,1},{0,1,0,1,0},{0,1,1,1,0},{1,0,0,0,1},{1,0,0,0,1},{1,1,1,1,1},{1,0,0,0,1},{0,1,1,1,0},{1,1,1,1,1},{1,0,1,0,1},{1,0,1,0,1},{1,1,1,1,1},{1,0,1,0,0},{1,0,1,0,0},{0,1,1,1,0},{1,0,0,0,1},{1,0,1,1,1},{1,1,1,1,1},{0,0,1,0,0},{1,1,1,1,1},{1,0,0,0,1},{1,1,1,1,1},{1,0,0,0,1},{0,0,0,1,1},{0,0,0,0,1},{1,1,1,1,1},{1,1,1,1,1},{0,0,1,0,0},{1,1,0,1,1},{1,1,1,1,1},{0,0,0,0,1},{0,0,0,0,1},{1,1,1,1,1},{0,1,0,0,0},{1,1,1,1,1},{1,1,1,1,1},{1,0,0,0,0},{1,1,1,1,1},{1,1,1,1,1},{1,0,0,0,1},{1,1,1,1,1},{1,1,1,1,1},{1,0,1,0,0},{1,1,1,0,0},{1,1,1,1,1},{1,1,1,1,1},{0,0,0,0,1},{1,1,1,1,1},{1,0,1,1,0},{1,1,1,0,1},{1,1,1,0,1},{1,0,1,0,1},{1,0,1,1,1},{1,0,0,0,0},{1,1,1,1,1},{1,0,0,0,0},{1,1,1,1,1},{0,0,0,0,1},{1,1,1,1,1},{1,1,1,1,1},{0,0,0,1,1},{1,1,1,1,0},{1,1,1,1,1},{0,0,0,1,0},{1,1,1,1,1},{1,1,0,1,1},{0,0,1,0,0},{1,1,0,1,1},{1,1,1,0,0},{0,0,1,1,1},{1,1,1,0,0},{1,0,0,1,1},{1,0,1,0,1},{1,1,0,0,1},{0,0,0,0,0},{1,1,1,1,1},{1,0,0,0,1},{1,1,0,0,0},{0,1,1,1,0},{0,0,0,1,1},{1,0,0,0,1},{1,1,1,1,1},{0,0,0,0,0},{0,1,0,0,0},{1,0,0,0,0},{0,1,0,0,0},{0,0,0,0,1},{0,0,0,0,1},{0,0,0,0,1},{0,0,0,0,0},{1,0,0,0,0},{0,1,0,0,0},{1,1,1,1,1},{1,0,1,0,0},{1,1,1,1,1},{1,1,1,1,1},{1,0,1,0,1},{0,1,0,1,0},{0,1,1,1,0},{1,0,0,0,1},{1,0,0,0,1},{1,1,1,1,1},{1,0,0,0,1},{0,1,1,1,0},{1,1,1,1,1},{1,0,1,0,1},{1,0,1,0,1},{1,1,1,1,1},{1,0,1,0,0},{1,0,1,0,0},{0,1,1,1,0},{1,0,0,0,1},{1,0,1,1,1},{1,1,1,1,1},{0,0,1,0,0},{1,1,1,1,1},{1,0,0,0,1},{1,1,1,1,1},{1,0,0,0,1},{0,0,0,1,1},{0,0,0,0,1},{1,1,1,1,1},{1,1,1,1,1},{0,0,1,0,0},{1,1,0,1,1},{1,1,1,1,1},{0,0,0,0,1},{0,0,0,0,1},{1,1,1,1,1},{0,1,0,0,0},{1,1,1,1,1},{1,1,1,1,1},{1,0,0,0,0},{1,1,1,1,1},{1,1,1,1,1},{1,0,0,0,1},{1,1,1,1,1},{1,1,1,1,1},{1,0,1,0,0},{1,1,1,0,0},{1,1,1,1,1},{1,1,1,1,1},{0,0,0,0,1},{1,1,1,1,1},{1,0,1,1,0},{1,1,1,0,1},{1,1,1,0,1},{1,0,1,0,1},{1,0,1,1,1},{1,0,0,0,0},{1,1,1,1,1},{1,0,0,0,0},{1,1,1,1,1},{0,0,0,0,1},{1,1,1,1,1},{1,1,1,1,1},{0,0,0,1,1},{1,1,1,1,0},{1,1,1,1,1},{0,0,0,1,0},{1,1,1,1,1},{1,1,0,1,1},{0,0,1,0,0},{1,1,0,1,1},{1,1,1,0,0},{0,0,1,1,1},{1,1,1,0,0},{1,0,0,1,1},{1,0,1,0,1},{1,1,0,0,1},{0,0,1,0,0},{1,1,1,1,1},{1,0,0,0,1},{0,0,0,0,0},{1,1,1,1,1},{0,0,0,0,0},{1,0,0,0,1},{1,1,1,1,1},{0,0,1,0,0},{0,0,1,0,0},{0,1,1,0,0},{0,0,0,0,0}};  #endif |
| HTTP.h |
| void httpSetup();  void updateWeather();  void refreshAPIs();  void updateTime();  void recconectWifi();  void recconectWifi()  {    if(WiFi.isConnected()==0)      myWIFI.reconectWIFI();  }   //timezonedb.com api  void updateTime()  {    if(WiFi.status()== WL\_CONNECTED)    {      HTTPClient http;      String timeApiPath="http://api.timezonedb.com/v2.1/get-time-zone?key="+timeApiKey+"&format=xml&by=position&lat="+(String)lat+"&lng="+(String)lng;      http.begin(timeApiPath.c\_str());      int httpResponseCode = http.GET();      if (httpResponseCode>0) {        Serial.print("HTTP Response code: ");        Serial.println(httpResponseCode);        String payload = http.getString();        int brojda1=payload.indexOf("<timestamp>")+strlen("<timestamp>");        int brojda2=payload.indexOf("</timestamp>");        //Serial.println(payload);        uint64\_t vrijemeSad=strtoul(payload.substring(brojda1,brojda2).c\_str(),nullptr,0);        Serial.println(vrijemeSad);        rtc.adjust(vrijemeSad);      }      else {        Serial.print("Error code: ");        Serial.println(httpResponseCode);      }      http.end();    }    else {      Serial.println("WiFi Disconnected");    }  }  //vrijemeZalaska vrijemeIzlaska  void updateWeather()  {    if(WiFi.status()== WL\_CONNECTED)    {      HTTPClient http;      String timeApiPath="https://api.openweathermap.org/data/2.5/weather?lat="+(String)lat+"&lon="+(String)lng+"&appid="+weatherApiKey+"&units=metric";      http.begin(timeApiPath.c\_str());      int httpResponseCode = http.GET();      if (httpResponseCode>0) {        Serial.print("HTTP Response code: ");        Serial.println(httpResponseCode);        String payload = http.getString();        DynamicJsonDocument jsonDoc(5096);        DeserializationError error = deserializeJson(jsonDoc, payload);        Serial.println(payload);        if (error) {          Serial.print(F("updateWeather() deserializeJson() failed with code "));          Serial.println(error.c\_str());        }        else        {          JsonObject root = jsonDoc.as<JsonObject>();          tempCurr=(float)root["main"]["temp"];          humCurr=(float)root["main"]["humidity"];          vrijemeIzlaska=(((int)root["sys"]["sunrise"]+(int)root["timezone"])%86400/60.0/60);          vrijemeZalaska=(((int)root["sys"]["sunset"]+(int)root["timezone"])%86400/60.0/60);          windCurr=(float)root["wind"]["speed"];          pressCurr=(int)root["main"]["pressure"];          visibilityCurr=(int)root["visibility"];          skyCurr=(int)root["clouds"]["all"];        }      }      else {        Serial.print("Error code: ");        Serial.println(httpResponseCode);      }      http.end();    }    else {      Serial.println("WiFi Disconnected");    }  }  void notFound(AsyncWebServerRequest \*request) {      request->send(404, "text/html", "<!doctype html><html><head><meta http-equiv=\"Content-Type\" content=\"text/html;charset=UTF-8\" /></head> <body>Stranica nije pronađena!<br> <a href=\"/\"> <input id=\"posaljitelj\" type=\"submit\" value=\"Početna stranica!\" style=\"height: 4em;  width: 10em;\"></a></body></html>");  }  void httpSetup()  {    HTTP.on("/", HTTP\_GET, [](AsyncWebServerRequest \*request){          request->send(SPIFFS, "/index.html", "text/html");    });    HTTP.on("/reset", HTTP\_GET, [](AsyncWebServerRequest \*request){      request->send(200,"text/html", "<!doctype html> <html><script>window.location.replace(\"/\");</script></html>");      delay(1500);      ESP.restart();    });    HTTP.on("/get", HTTP\_GET, [] (AsyncWebServerRequest \*request) {      String tmp;      bool changedAny=0;      if (request->hasParam("latitude") && request->getParam("latitudechange")->value()=="1") {        lat=atof((request->getParam("latitude")->value()).c\_str());        changedAny=1;      }      if (request->hasParam("longitude") && request->getParam("longitudechange")->value()=="1") {        lng=atof((request->getParam("longitude")->value()).c\_str());        changedAny=1;      }      if (request->hasParam("boja1") && request->getParam("boja1change")->value()=="1") {        tmp=(request->getParam("boja1")->value());        tmp = tmp.substring(1, tmp.length() );        int r,g,b;        sscanf(tmp.c\_str(), "%02x%02x%02x", &r, &g, &b);        boja1[0]=r;boja1[1]=g;boja1[2]=b;        changedAny=1;      }      if (request->hasParam("boja2") && request->getParam("boja2change")->value()=="1") {        tmp=request->getParam("boja2")->value();        tmp = tmp.substring(1, tmp.length() );        int r,g,b;        sscanf(tmp.c\_str(), "%02x%02x%02x", &r, &g, &b);        boja2[0]=r;boja2[1]=g;boja2[2]=b;        changedAny=1;      }      if (request->hasParam("boja3") && request->getParam("boja3change")->value()=="1") {        tmp=request->getParam("boja3")->value();        tmp = tmp.substring(1, tmp.length() );        int r,g,b;        sscanf(tmp.c\_str(), "%02x%02x%02x", &r, &g, &b);        boja3[0]=r;boja3[1]=g;boja3[2]=b;        changedAny=1;      }      if (request->hasParam("SSID") && request->getParam("SSIDchange")->value()=="1")  {        SSID=request->getParam("SSID")->value();        changedAny=1;      }      if (request->hasParam("PASSWORD") && request->getParam("PASSWORDchange")->value()=="1") {        PASS=request->getParam("PASSWORD")->value();        changedAny=1;      }      if (request->hasParam("APSSID") && request->getParam("APSSIDchange")->value()=="1") {        APssid=request->getParam("APSSID")->value();        changedAny=1;      }      if (request->hasParam("APPASSWORD") && request->getParam("APPASSWORDchange")->value()=="1") {        APpass=request->getParam("APPASSWORD")->value();        changedAny=1;      }      if (request->hasParam("timezonedb") && request->getParam("timezonedbchange")->value()=="1") {        timeApiKey=request->getParam("timezonedb")->value();        changedAny=1;      }      if (request->hasParam("openweathermap") && request->getParam("openweathermapchange")->value()=="1") {        weatherApiKey=request->getParam("openweathermap")->value();        changedAny=1;      }      if (request->hasParam("mod") && request->getParam("modchange")->value()=="1") {        stateID=(request->getParam("mod")->value()).toInt();        changedAny=1;      }      if(changedAny)      {        saveConfig();        refreshAPIs();        myWIFI.setConfigWIFI(SSID.c\_str(), PASS.c\_str(), APssid.c\_str(), APssid.c\_str(), APpass.c\_str());      }      tmp="<!doctype html><head><meta http-equiv=\"Content-Type\" content=\"text/html;charset=UTF-8\" /></head>";      tmp+="<style>input[type=button]{width: 75%; height: 1.6em;}";      tmp+="html{background-color: #DBDBDB;color: #376DAB;font-family: monospace;font-weight: bolder;}";      tmp+="input:hover{transition: 0.5s; border: 2px brown solid;}input:focus{background-color: yellow;}";      tmp+="input{font-size: 2vw;height: 1.5em;padding: auto;margin: 0.1em;color: #376DAB; font-family: monospace;background-color: #EE635240;border: 2px transparent solid;border-radius: 0.2em;font-weight: bolder;position: relative; width: 100%; transition: 0.5s;padding: 0px;height: 1.6em;}</style>";      tmp+="<body><div style=\"display: flex; flex-direction: column;\"><h1>Poslano!</h1> <br> <a href=\"/\"> <input id=\"posaljitelj\" type=\"submit\" value=\"Vrati se nazad!\" style=\"height: 3em;  width: 99%;\"></a>";      tmp+="<br><br><br> <form action=\"/reset\"> <a href=\"/\"> <input id=\"posaljitelj\" type=\"submit\" value=\"Restart Sata? (tako se wifi ponovo spaja na novi)\" style=\"height: 3em;  width: 99%;\"></a></form><br> </div><body>";      //debug!      if(1==0)      {      tmp+="<table>";      tmp+="<tr><th>Varijabla</th><th>Vrijednost</th></tr>";      tmp+="<tr><td>Geografska Širina</td><td>"+(String)lat+"</td></tr>";      tmp+="<tr><td>Geografska Dužina</td><td>"+(String)lng+"</td></tr>";      tmp+="<tr><td>Boja1</td><td>("+(String)boja1[0]+", "+(String)boja1[1]+", "+(String)boja1[2]+")</td></tr>";      tmp+="<tr><td>Boja2</td><td>("+(String)boja2[0]+", "+(String)boja2[1]+", "+(String)boja2[2]+")</td></tr>";      tmp+="<tr><td>Boja3</td><td>("+(String)boja3[0]+", "+(String)boja3[1]+", "+(String)boja3[2]+")</td></tr>";      tmp+="<tr><td>WiFi SSID</td><td>"+(String)myWIFI.getNameSSID()+"</td></tr>";      tmp+="<tr><td>WiFi Šifra</td><td>"+(String)myWIFI.getPassSSID()+"</td></tr>";      tmp+="<tr><td>HotSpot SSID</td><td>"+(String)myWIFI.getNameAPSSID()+"</td></tr>";      tmp+="<tr><td>HotSpot Šifra</td><td>"+(String)myWIFI.getPassAPSSID()+"</td></tr>";      tmp+="<tr><td>Timezonedb Api</td><td>"+(String)timeApiKey+"</td></tr>";      tmp+="<tr><td>openweathermap Api</td><td>"+(String)weatherApiKey+"</td></tr>";      tmp+="<tr><td>ssdp</td><td>"+(String)myWIFI.getNameSSDP()+"</td></tr>";      tmp+="</table>";      }      request->send(200,"text/html", tmp);      });    HTTP.onNotFound(notFound);    HTTP.begin();    }  void refreshAPIs()  {      updateTime();      updateWeather();  } |
| display.h |
| void clearAllLayers();  void drawBroj(int x, int y, int znak, int scaling, uint8\_t \*boje);  void drawClock(int x, int y, int scaling, uint8\_t \*boje);  void drawLetter(int x, int y, char znak, uint8\_t \*boje);  void typeText(int x, int y, String text, uint8\_t \*boje);  void drawDaytimeLine(int x1, int y1, int x2, int y2, float sat, int povecanje);  void typeTextNoNl(int x, int y, String text, uint8\_t \*boje);  void drawLetterNoNl(int x, int y, char znak, uint8\_t \*boje);  void refreshTime();  void swapAllLayerBuffers();  void clearAllLayerBuffers();  bool gumbPress(uint8\_t ulaz);  void gumbiSense();  void displayDraw();  void drawDaytimeLine(int x1, int y1, int x2, int y2, float sat, int povecanje=60)  {        if (vrijemeZalaska<vrijemeIzlaska)vrijemeZalaska+=24;        int satx=0;        for(int i=x1; i<x2; i++)        {          satx=i/64.0\*24+sat;          if(satx>=24) satx-=24;          if(satx>vrijemeIzlaska and satx<vrijemeZalaska)          {            if(dan!=255)            {              dan+=(dan+povecanje>255?255%povecanje-dan%povecanje:povecanje);              noc-=(noc-povecanje<0?255%povecanje-dan%povecanje:povecanje);            }          }          else          {            if(noc!=255)            {              noc+=(noc+povecanje>255?255%povecanje-noc%povecanje:povecanje);              dan-=(dan-povecanje<0?255%povecanje-dan%povecanje:povecanje);            }          }          backgroundLayer.drawFastVLine(i, y1, y2,{dan,dan,noc});        }  }  void drawLetter(int x, int y, char znak, uint8\_t \*boje)  {    int trenutnoSlovo=((int)znak)-32;    for(int j=0 ;j<5; j++)    {      int pomBr=0;        for(int i=trenutnoSlovo\*3; i<3+trenutnoSlovo\*3; i++)        {            int brojkeX=i;            int ypos=y+j;            int xpos=x+pomBr++;            backgroundLayer.drawPixel(xpos, ypos, {\*(boje)\*miniLetters[brojkeX][j] , \*(boje+1)\*miniLetters[brojkeX][j], \*(boje+2)\*miniLetters[brojkeX][j]});        }    }  }  void typeText(int x, int y, String text, uint8\_t \*boje)  {    int xpoc=x;    for(int i=0; i<text.length(); i++)    {      if((char)text[i]=='\n'){y=y+6; x=xpoc; i+=1; if(i>=text.length()) return;}        drawLetter(x, y, (char)text[i], boje);      if(x+6<kMatrixWidth) x+=4; else if(y+11<kMatrixHeight) {y=y+6; x=xpoc;} else return;      }  }  void drawLetterNoNl(int x, int y, char znak, uint8\_t \*boje)  {    int trenutnoSlovo=((int)znak)-32;    for(int j=0 ;j<5; j++)    {      int pomBr=0;        for(int i=trenutnoSlovo\*3; i<3+trenutnoSlovo\*3; i++)        {            int brojkeX=i;            int ypos=y+j;            int xpos=x+pomBr++;            if(x>=0 and x<=kMatrixWidth)backgroundLayer.drawPixel(xpos, ypos, {\*(boje)\*miniLetters[brojkeX][j] , \*(boje+1)\*miniLetters[brojkeX][j], \*(boje+2)\*miniLetters[brojkeX][j]});        }    }  }  void typeTextNoNl(int x, int y, String text, uint8\_t \*boje)  {    int xpoc=x;    for(int i=0; i<text.length(); i++)    {      drawLetterNoNl(x, y, (char)text[i], boje);      x+=4;    }  }  void drawBroj(int x, int y, int znak, int scaling, uint8\_t \*boje)  {      for(int j=0; j<10; j++)      {        int pomBr=0;          for(int i=znak\*5; i<5+znak\*5; i++)          {              int brojkeX=i;              int ypos=y+j\*scaling;              int xpos=x+pomBr;              pomBr+=scaling;              for(int i1=0; i1<scaling; i1++)for(int j1=0; j1<scaling; j1++)                backgroundLayer.drawPixel(xpos+i1, ypos+j1, {\*(boje)\*brojke[brojkeX][j] , \*(boje+1)\*brojke[brojkeX][j], \*(boje+2)\*brojke[brojkeX][j]});          }      }  }  void drawClock(int x, int y, int scaling, uint8\_t \*boje)  {        String vrijemeIspisa=(String)((sat-sat%10)/10)+ (String)(sat%10)+(tocke ? ":":" ")+(String)((minuta-minuta%10)/10)+ (String)(minuta%10);        int tockaMinus=0;      for(int i=0;i<vrijemeIspisa.length();i++)        if(vrijemeIspisa[i]==':' and tocke)          {drawBroj((x+i\*6\*scaling+tockaMinus), y, 11, scaling, boje); tockaMinus-=scaling;}        else        if(vrijemeIspisa[i]!=' ' and vrijemeIspisa[i]!=':')          drawBroj((x+i\*6\*scaling+tockaMinus), y, ((String)vrijemeIspisa[i]).toInt(), scaling, boje);        else          tockaMinus-=scaling;  //Serial.println("!da!");  //Serial.println(tocke);  }  void clearAllLayers() {    backgroundLayer.fillScreen((rgb24)BLACK);    backgroundLayer.swapBuffers();    indexedLayerZ1.fillScreen(BLACK);    indexedLayerZ1.swapBuffers();    indexedLayerZ2.fillScreen(BLACK);    indexedLayerZ2.swapBuffers();  }  void clearAllLayerBuffers() {    backgroundLayer.fillScreen((rgb24)BLACK);    indexedLayerZ1.fillScreen(BLACK);    indexedLayerZ2.fillScreen(BLACK);  }  void swapAllLayerBuffers() {    backgroundLayer.swapBuffers();    indexedLayerZ1.swapBuffers();    indexedLayerZ2.swapBuffers();  }  void refreshTime()  {      DateTime now = rtc.now();    sat=now.hour();    minuta=now.minute();    sekunda=now.second();      tocke=!tocke;  }  bool gumbPress(uint8\_t ulaz)  {    adc0.setMultiplexer(ulaz);    if(adc0.getMilliVolts(false)>1000)      return true;    else      return false;  }  void gumbiSense()  {    if(++zadnje\_stisnuto>2) //svakih cca 0.5s    {      gumbStanje=0;    }    bool g0=gumbPress(ADS1115\_MUX\_P0\_NG);    bool g1=gumbPress(ADS1115\_MUX\_P1\_NG);    bool g2=gumbPress(ADS1115\_MUX\_P2\_NG);    bool g3=gumbPress(ADS1115\_MUX\_P3\_NG);    bool jelStisnuto=(g0 or g1 or g2 or g3);    if(gumbStanje==0 and jelStisnuto)    {      gumbStanje=1;      if(g0)      {        stateID+=1;        if(stateID>4) stateID=1;        saveStanje=1;      }      else if(g1)      {        if (svjetlina+50>255) svjetlina=50; else svjetlina+=50;        matrix.setBrightness(svjetlina);        saveStanje=1;      }      else if(g2)      {        uint8\_t r= boja1[0],g= boja1[1],b= boja1[2];        boja1[0]=boja2[0];    boja1[1]=boja2[1];    boja1[2]=boja2[2];        boja2[0]=boja3[0];    boja2[1]=boja3[1];    boja2[2]=boja3[2];        boja3[0]=r;           boja3[1]=g;           boja3[2]=b;        saveStanje=1;      }      else if(g3)      {        clearAllLayers();        indexedLayerZ1.drawString(0,0,1,"Wi-Fi detalji");        if(WiFi.status() == WL\_CONNECTED)          typeText(0,7,(String)("SSID:\n"+(String)myWIFI.getNameSSID()+"\nIP-WiFi:\n"+myWIFI.getDevStatusIP().substring(9)),&boja2[0]);        else          typeText(0,7,(String)("SSID:\n"+APssid+"\nIP-HOTSPOT:\n192.168.4.1"),&boja2[0]);        swapAllLayerBuffers();        delay(5000);      }      zadnje\_stisnuto=0;    }    if(saveStanje==1)      if(zadnje\_stisnuto>50 ) //nakon cca 10s      {        saveStanje=0;          saveConfig();      }  }  void displayDraw()  {    matrix.setBrightness(svjetlina);    clearAllLayerBuffers();    if(stateID==1)    {      statusTrenutno= (String)tempCurr + "\*C-temp.   "+ (String)humCurr+"%RH-vlaga   " + (String)windCurr+"m/s-vjetar   "+ (String)pressCurr+"hPa-tlak   " +(String) visibilityCurr+"m-vidljivost   "+ (String)skyCurr + "%-oblaka   ";      drawClock(4,1,2, &boja1[0]);      drawDaytimeLine(0,22,64,23,(sat+minuta/60.0),50);      typeTextNoNl(--statusTrenutniPomak,25,statusTrenutno,&boja2[0]);      // Serial.println(4\*da.length());      //Serial.println(xda>4\*da.length());      // Serial.println(xda);      if(-(int)statusTrenutniPomak>(int)(4\*statusTrenutno.length())) {statusTrenutniPomak=kMatrixWidth;}        // Serial.println(xda);    }    else if(stateID==2)    {      drawClock(4,1,2, &boja1[0]);      drawDaytimeLine(0,22,64,23,(sat+minuta/60.0),50);      String datum=(String)rtc.now().day()+"."+(String)rtc.now().month()+"."+(String)rtc.now().year()+".";      typeTextNoNl((16-datum.length())/2.0\*4,25, datum,&boja2[0]);    }    else if(stateID==3)    {      drawClock(4,1,2, &boja1[0]);      String datum=(String)rtc.now().day()+"."+(String)rtc.now().month()+"."+(String)rtc.now().year()+".";      typeTextNoNl((16-datum.length())/2.0\*4,25, datum,&boja2[0]);    }    else if(stateID==4)    {      statusTrenutno= (String)tempCurr + "\*C-temp.   "+ (String)humCurr+"%RH-vlaga   " + (String)windCurr+"m/s-vjetar   "+ (String)pressCurr+"hPa-tlak   " +(String) visibilityCurr+"m-vidljivost   "+ (String)skyCurr + "%-oblaka   ";      drawClock(4,1,2, &boja1[0]);      drawDaytimeLine(0,22,64,23,(sat+minuta/60.0),50);      typeTextNoNl(--statusTrenutniPomak,25,statusTrenutno,&boja2[0]);      // Serial.println(4\*da.length());      //Serial.println(xda>4\*da.length());      // Serial.println(xda);      if(-(int)statusTrenutniPomak>(int)(4\*statusTrenutno.length())) {statusTrenutniPomak=kMatrixWidth;}        // Serial.println(xda);      if(WiFi.status() == WL\_CONNECTED)        backgroundLayer.drawFastHLine(15,48,31,{0,64,128});      else        backgroundLayer.drawFastHLine(15,48,31,{128,64,0});    }    swapAllLayerB uffers();  } |
| HTML geolokacija |
| <!DOCTYPE html>  <html>  <style>    html {      background-color: #DBDBDB;      color: #376DAB;      font-family: monospace;      font-weight: bolder;    }    table,    th,    td {      border: 1px solid #C14953;    }    input,    select {      height: 1.5em;      padding: auto;      margin: 0.1em;      color: #376DAB;      font-family: monospace;      background-color: lightblue;      border: 2px transparent solid;      border-radius: 0.2em;      font-weight: bolder;      position: relative;      width: 100%;      transition: 0.5s;      padding: 0px;      height: 1.6em;    }    input[type=color],    input[type=button],    select {      width: 75%;      height: 1.6em;    }    input:hover {      transition: 0.5s;      border: 2px brown solid;    }    input:focus {      background-color: yellow;    }    label {      height: 2em;    }    h3 {      margin-top: 2px;    }    section {      border-radius: 1em;      border: 3px solid #C14953;      background-color: #30292F;      padding: 1em;      margin: 2px;    }    a,    button {      color: #376DAB;      border: solid 0.2em gray;      padding: 0.5em;      margin: 0.25em;      border-radius: 0.5em;      background-color: gainsboro;      transition: 0.25s;      -webkit-user-select: none;      -ms-user-select: none;      user-select: none;      font-family: monospace;      font-weight: bolder;    }    a:hover{      color: #4b91e0;      background-color: lightgray;      transition: 0.25s;    }    a:active {      background-color: yellow;      transition: 0.25s;    }  </style>  <head>    <meta http-equiv="Content-Type" content="text/html;charset=UTF-8" />  </head>  <body>    <div id="forma"  style="left: 25%; position: absolute; width: 50%;">      <h1 style="margin-bottom: 0px;">        Digitalni sat      </h1>      <h3 style="margin-top: 0px;">(Završni rad Niko Pešut)</h3>      <section style="background-color:rgba(128, 128, 128, 0.2); border-color: gray;">          <h3>Stisni da bi dobio/la lokaciju</h3>        <a onclick="getLocation()">Stisni me</a>        <div id="izlaz"></div>      </section>      </div>    </div>    <script>      var x = document.getElementById("izlaz");      function getLocation() {        if (navigator.geolocation) {          navigator.geolocation.getCurrentPosition(showPosition);        } else {          x.innerHTML = "Geolokacija nije podržana u ovom pregledniku!";        }      }      function showPosition(position) {        x.innerHTML = "<h3><br> Geografska Širina: <input type=\"text\" value=\"" + position.coords.latitude +          "\" id=\"sirina\"><br>Geografska Dužina: <input type=\"text\" value=\"" + position.coords.longitude + "\" id=\"duzina\"><br><br>Sada te podatke prekopiraj u lokalni form/obrazac.</h3>";        x.innerHTML += "Kopiraj klikom:<br><br><a onclick=\"gsir()\">Geografska Širina</a><a onclick=\"gduz()\">Geografska Dužina</a>";      }      function gduz() {        var copyText = document.getElementById("duzina");        copyText.select();        copyText.setSelectionRange(0, 99999);        navigator.clipboard.writeText(copyText.value);      }      function gsir() {        var copyText = document.getElementById("sirina");        copyText.select();        copyText.setSelectionRange(0, 99999);        navigator.clipboard.writeText(copyText.value);      }    </script>  </body>  </html> |
| HTML konfiguracija |
| <!DOCTYPE html>  <html>  <style>    html {      background-color: #DBDBDB;      color: #376DAB;      font-family: monospace;      font-weight: bolder;    }    table,    th,    td {      border: 1px solid #C14953;    }    input,    select {      height: 1.5em;      padding: auto;      margin: 0.1em;      color: #376DAB;      font-family: monospace;      background-color: #EE635240;      border: 2px transparent solid;      border-radius: 0.2em;      font-weight: bolder;      position: relative;      width: 100%;      transition: 0.5s;      padding: 0px;      height: 1.6em;    }    input[type=color],    input[type=button],    select {      width: 75%;      height: 1.6em;    }    input:hover {      transition: 0.5s;      border: 2px brown solid;    }    input:focus {      background-color: yellow;    }    label {      height: 2em;    }    h3 {      margin-top: 2px;    }    section {      border-radius: 1em;      border: 3px solid #C14953;      background-color: #30292F;      padding: 1em;      margin: 2px;    }    a {      color: #376DAB;    }  </style>  <head>    <meta http-equiv="Content-Type" content="text/html;charset=UTF-8" />  </head>  <body>    <form id="forma" action="/get" method="get" style="left: 25%; position: absolute; width: 50%;">      <div style="display: flex; flex-direction: column;">        <h1 style="margin-bottom: 0px;">          Digitalni sat        </h1>        <h3 style="margin-top: 0px;">(Završni rad Niko Pešut)</h3>        <section style="background-color:rgba(128, 128, 128, 0.2); border-color: gray;">          <h3 style="margin: 0px;">Stavke koje ne mijenjate se neće postaviti</h3>        </section>        <section>          <h2 style="margin-bottom: 2px; ">Lokacija</h2>          <h5 style="margin-top: 2px;">(od lokacije se automatski uzima vremenska zona, datum, vrijeme...)</h1>            <a href="https://geoloc.pesut.win" target="\_blank"><input type="button"  value="dobij lokaciju"></a><br>            <label for="latitude">Geografska Širina</label>            <input type="text" id="latitude" name="latitude"><br>            <label for="longitude">Geografska Dužina</label>            <input type="text" id="longitude" name="longitude"><br>            <input type="hidden" id="longitudechange" name="longitudechange" value="0">            <input type="hidden" id="latitudechange" name="latitudechange" value="0">        </section>        <section>          <h2>Boje i stil</h2>          <label for="boja1">1. boja</label><br>          <input type="color" value="#ffffff" name="boja1" id="boja1"><br>          <label for="boja2">2. boja</label><br>          <input type="color" value="#ffffff" name="boja2" id="boja2"><br>          <label for="boja3">3. boja</label><br>          <input type="color" value="#ffffff" name="boja3" id="boja3"><br>          <label for="mod">Modovi</label><br>          <select name="mod" id="mod">            <option value="0" disabled selected>-- Odaberi mod --</option>            <option value="4">Sat, crta dana, trenutačno vrijeme i wifi status</option>            <option value="1">Sat, crta dana i trenutačno vrijeme</option>            <option value="2">Sat, crta dana i datum</option>            <option value="3">Sat i datum</option>          </select>          <input type="hidden" id="boja1change" name="boja1change" value="0">          <input type="hidden" id="boja2change" name="boja2change" value="0">          <input type="hidden" id="boja3change" name="boja3change" value="0">          <input type="hidden" id="modchange" name="modchange" value="0">        </section>        <section id="mreza">          <h2>Mreža</h2>          <label for="SSID">WiFi SSID</label>          <input id="SSID" name="SSID" type="text" placeholder="Ime WIFI-a"><br>          <label for="PASSWORD">WiFi Šifra</label>          <input id="PASSWORD" name="PASSWORD" type="text" placeholder="Šifra WIFI-a"><br>          <label for="APSSID">HotSpot SSID</label>          <input id="APSSID" name="APSSID" type="text" placeholder="Ime accespoint-a"><br>          <label for="APPASSWORD">HotSpot Šifra</label>          <input id="APPASSWORD" name="APPASSWORD" type="text" placeholder="Šifra accespoint-a"><br>          <input type="hidden" id="SSIDchange" name="SSIDchange" value="0">          <input type="hidden" id="PASSWORDchange" name="PASSWORDchange" value="0">          <input type="hidden" id="APSSIDchange" name="APSSIDchange" value="0">          <input type="hidden" id="APPASSWORDchange" name="APPASSWORDchange" value="0">        </section>        <section>          <h2>API-s</h2>          <label for="timezonedb"><a href="https://www.timezonedb.com">Timezonedb</a> api ključ</label>          <input id="timezonedb" name="timezonedb" type="text" placeholder="Api Key"><br>          <label for="openweathermap"><a href="https://www.openweathermap.org">Openweathermap</a> api ključ</label>          <input id="openweathermap" name="openweathermap" type="text" placeholder="Api Key"><br>          <input type="hidden" id="timezonedbchange" name="timezonedbchange" value="0">          <input type="hidden" id="openweathermapchange" name="openweathermapchange" value="0">        </section>        <input type="submit" id="posaljitelj" type="button" value="Pošalji" style="height: 3em; font-size: 2em; width: 99%;">      </div>    </form>  </body>  <script>    document.getElementById("boja1change").value = "0";    document.getElementById("boja2change").value = "0";    document.getElementById("boja3change").value = "0";    document.getElementById("latitudechange").value = "0";    document.getElementById("longitudechange").value = "0";    document.getElementById("SSIDchange").value = "0";    document.getElementById("PASSWORDchange").value = "0";    document.getElementById("APSSIDchange").value = "0";    document.getElementById("APPASSWORDchange").value = "0";    document.getElementById("timezonedbchange").value = "0";    document.getElementById("openweathermapchange").value = "0";    document.getElementById("modchange").value = "0";    document.getElementById("boja1").addEventListener("change", (event) => { document.getElementById("boja1change").value = "1"; });    document.getElementById("boja2").addEventListener("change", (event) => { document.getElementById("boja2change").value = "1"; });    document.getElementById("boja3").addEventListener("change", (event) => { document.getElementById("boja3change").value = "1"; });    document.getElementById("latitude").addEventListener("change", (event) => { document.getElementById("latitudechange").value = "1"; });    document.getElementById("longitude").addEventListener("change", (event) => { document.getElementById("longitudechange").value = "1"; });    document.getElementById("SSID").addEventListener("change", (event) => { document.getElementById("SSIDchange").value = "1"; });    document.getElementById("PASSWORD").addEventListener("change", (event) => { document.getElementById("PASSWORDchange").value = "1"; });    document.getElementById("APSSID").addEventListener("change", (event) => { document.getElementById("APSSIDchange").value = "1"; });    document.getElementById("APPASSWORD").addEventListener("change", (event) => { document.getElementById("APPASSWORDchange").value = "1"; });    document.getElementById("timezonedb").addEventListener("change", (event) => { document.getElementById("timezonedbchange").value = "1"; });    document.getElementById("openweathermap").addEventListener("change", (event) => { document.getElementById("openweathermapchange").value = "1"; });    document.getElementById("mod").addEventListener("change", (event) => { document.getElementById("modchange").value = "1"; });    let form = document.getElementById("forma");    let button = document.getElementById("get-location");    let latText = document.getElementById("latitude");    let longText = document.getElementById("longitude");    document.getElementById("get-location").addEventListener("click", (event) => {      navigator.geolocation.getCurrentPosition((position) => {        let lat = position.coords.latitude;        let long = position.coords.longitude;        latText.value = lat.toFixed(2);        longText.value = long.toFixed(2);      }      );    }    );    document.getElementById("posaljitelj").addEventListener("click", (event) => {      form.submit();    }    );  </script>  </html> |